

# Exemplar Candidate Work

Part 2 of 2

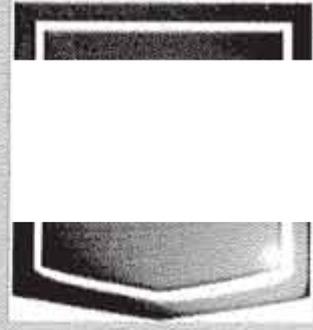
## **GCE in Applied ICT**

OCR Advanced GCE in Applied ICT: H715

Unit G057: Database design

# Menus for End User

# Technology College Library - Main Menu



Reports

About

Exit

## Edit Details:

Student Loans

Books

Students

Form Groups

18 March 2010

# [REDACTED] Database System

Designed and created by [REDACTED]

Version Number: 1.2.2

(Created 8th October 2009, Updated 18th March 2010)

This Database was Designed on Microsoft Access 2007

For technical help please refer to your user guide or Call our technical support team on 01294-546798

Copyright © [REDACTED] All Rights Reserved.

Table: Books

Page: 1

*The following section details the 'books' table.*

**Properties**

DateCreated:	05/07/2009 22:33:18	DefaultView:	2
DisplayViewsOnSharePointSit	1	FilterOnLoad:	False
GUID:	{guid {384B2729-FF39-4822-876C-E219D01AD294}}	HideNewField:	False
LastUpdated:	18/03/2010 14:44:19	NameMap:	Long binary data
OrderByOn:	False	OrderByOnLoad:	True
Orientation:	Left-to-Right	RecordCount:	30
TotalsRow:	False	Updatable:	True

**Columns**

Name	Type	Size
<b>Book ID</b>	Long Integer	4
AggregateType:	-1	
AllowZeroLength:	False	
AppendOnly:	False	
Attributes:	Fixed Size, Auto-Increment	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	1	
ColumnWidth:	Default	
DataUpdatable:	False	
GUID:	{guid {B859F844-963F-40B0-99F4-BC2AC032D908}}	
OrdinalPosition:	0	
Required:	False	
SourceField:	Book ID	
SourceTable:	Books	
TextAlign:	General	
<b>ISBN Number</b>	Text	255
AggregateType:	-1	
AllowZeroLength:	False	
AppendOnly:	False	
Attributes:	Variable Length	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	2	
ColumnWidth:	2055	
DataUpdatable:	False	
DisplayControl:	Text Box	
GUID:	{guid {4C3A1CD3-CFC7-40F2-87B1-9DB7DE6C912E}}	
IMEMode:	0	
IMESentenceMode:	3	
InputMask:	000\0\00000000\00\0	
OrdinalPosition:	1	
Required:	True	
SourceField:	ISBN Number	
SourceTable:	Books	
TextAlign:	General	
UnicodeCompression:	False	
<b>Book Name</b>	Text	30

*The book ID is used as the primary key and is an auto-number field.*

*Input mask sets the format of data*

*Field has to be completed*

Table: Books

Page: 2

AggregateType:	-1	
AllowZeroLength:	False	- Cannot be left empty
AppendOnly:	False	
Attributes:	Variable Length	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	3	
ColumnWidth:	2895	
DataUpdatable:	False	
DisplayControl:	Text Box	
GUID:	{guid {B6902D4D-88A4-4EEB-BF80-65131E9C2EB4}}	
IMEMode:	0	
IMESentenceMode:	3	
OrdinalPosition:	2	
Required:	True	- Name of book is required. No validation.
SourceField:	Book Name	
SourceTable:	Books	
TextAlign:	General	
UnicodeCompression:	True	

Author

Text

20

AggregateType:	-1	
AllowZeroLength:	False	- Cannot be left empty.
AppendOnly:	False	
Attributes:	Variable Length	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	4	
ColumnWidth:	1560	
DataUpdatable:	False	
DisplayControl:	Text Box	
GUID:	{guid {6D785648-D0B0-4C24-953E-D93FA1CF6C9C}}	
IMEMode:	0	
IMESentenceMode:	3	
OrdinalPosition:	3	
Required:	True	- field required to create record
SourceField:	Author	
SourceTable:	Books	
TextAlign:	General	
UnicodeCompression:	True	

Publisher

Text

20

AggregateType:	-1	
AllowZeroLength:	False	- field cannot be empty
AppendOnly:	False	
Attributes:	Variable Length	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	5	
ColumnWidth:	1635	
DataUpdatable:	False	
DisplayControl:	Text Box	
GUID:	{guid {5B198747-E823-4035-A353-AEC85A9B7B61}}	
IMEMode:	0	
IMESentenceMode:	3	
OrdinalPosition:	4	

Required:	True	- field is required
SourceField:	Publisher	
SourceTable:	Books	
TextAlign:	General	
UnicodeCompression:	True	

Genre

Text 20

AggregateType:	-1	
AllowZeroLength:	False	- field cannot be left empty
AppendOnly:	False	
Attributes:	Variable Length	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	6	
ColumnWidth:	2010	
DataUpdatable:	False	
DisplayControl:	Text Box	
GUID:	{guid {597BD65E-113E-4FF0-93E1-D085B77E84FB}}	
IMEMode:	0	
IMESentenceMode:	3	
OrdinalPosition:	5	
Required:	True	- field must be completed
SourceField:	Genre	
SourceTable:	Books	
TextAlign:	General	
UnicodeCompression:	True	

Price

Currency 8

AggregateType:	-1	
AllowZeroLength:	False	- Cannot be empty
AppendOnly:	False	
Attributes:	Fixed Size	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	7	
ColumnWidth:	Default	
DataUpdatable:	False	
DecimalPlaces:	2	
Format:	E#,##0.00;-E#,##0.00	Sets format and currency.
GUID:	{guid {0CA3C44C-7111-4151-8891-6A8DBAA78F86}}	
OrdinalPosition:	6	
Required:	True	- field required to enter record
SourceField:	Price	
SourceTable:	Books	
TextAlign:	General	
ValidationRule:	>0	- The price must be more than £0.00
ValidationText:	Book Price Cannot be equal to or less than £0.00, please re-enter value.	

Relationships

! Text shown if error occurs because of validation.

**BooksLoans**



Attributes: Enforced  
 RelationshipType: One-To-Many

*One-to-many relationship between Books table and Loans table.*

Table Indexes

Name	Number of Fields
ISBN Number	1
Clustered:	False
DistinctCount:	21
Foreign:	False
IgnoreNulls:	False
Name:	ISBN Number
Primary:	False
Required:	False
Unique:	False
Fields:	
ISBN Number	Ascending
PrimaryKey	1
Clustered:	False
DistinctCount:	30
Foreign:	False
IgnoreNulls:	False
Name:	PrimaryKey
Primary:	True
Required:	True
Unique:	True
Fields:	
Book ID	Ascending

*'Book ID' is primary key in Ascending order.*

The following pages show properties for the 'Form Groups' Table.

**Properties**

DateCreated:	05/07/2009 22:47:30	DefaultView:	2
DisplayViewsOnSharePointSite:	1	FilterOnLoad:	False
GUID:	{guid {C1D0CCFC-113D-45CF-BC1B-E46917D35438}}	HideNewField:	False
LastUpdated:	18/03/2010 13:32:42	NameMap:	Long binary data
OrderByOn:	False	OrderByOnLoad:	True
Orientation:	Left-to-Right	RecordCount:	12
TotalsRow:	False	Updatable:	True

**Columns**

Name	Type	Size
<b>Form ID</b>	Long Integer	4
AggregateType:	-1	
AllowZeroLength:	False	
AppendOnly:	False	
Attributes:	Fixed Size, Auto-Increment	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	1	
ColumnWidth:	Default	
DataUpdatable:	False	
GUID:	{guid {EE478BA3-C27E-44D9-A07B-8FFED0A891FA}}	
OrdinalPosition:	0	
Required:	False	
SourceField:	Form ID	
SourceTable:	Forms	
TextAlign:	General	

- Automatically increasing  
 - 'Form ID' is the primary key.

<b>Year Group</b>	Double	8
AggregateType:	-1	
AllowZeroLength:	False	
AppendOnly:	False	
Attributes:	Fixed Size	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	Default	
ColumnWidth:	Default	
DataUpdatable:	False	
DecimalPlaces:	Auto	
DisplayControl:	Text Box	
GUID:	{guid {C5C331D4-B424-4833-BBF4-B4786C891DBA}}	
OrdinalPosition:	1	
Required:	True	
SourceField:	Year Group	
SourceTable:	Forms	
TextAlign:	General	
ValidationRule:	Between 7 And 13	
ValidationText:	Year group must be between year 7 and year 13.	

- field cannot be left blank.

- field is required  
 - Validation means 'year group' must be 7, 8, 9, 10, 11, 12 or 13.

<b>Form Room</b>	Text	
AggregateType:	-1	

Error shown if validation not met.

Table: Forms

Page: 6

AllowZeroLength:	True
AppendOnly:	False
Attributes:	Variable Length
CollatingOrder:	General
ColumnHidden:	False
ColumnOrder:	Default
ColumnWidth:	Default
DataUpdatable:	False
DisplayControl:	Text Box
GUID:	{guid {49AFBCF9-9545-4418-89DB-51353541F529}}
IMEMode:	0
IMESentenceMode:	3
InputMask:	??##
OrdinalPosition:	2
Required:	True
SourceField:	Form Room
SourceTable:	Forms
TextAlign:	General
UnicodeCompression:	True
ValidationRule:	Like "TA*" Or Like "TB*" Or Like "TC*"
ValidationText:	Form Rooms must begin with TA, TB, or TC and must be followed by room number E.G. TB27

- field must be two letters followed by two numbers.

- Field has to be complete.

← First two letters must be TA, TB or TC.

- Text shown if validation not met.

Form Tutor

AggregateType:	-1
AllowZeroLength:	True
AppendOnly:	False
Attributes:	Variable Length
CollatingOrder:	General
ColumnHidden:	False
ColumnOrder:	Default
ColumnWidth:	Default
DataUpdatable:	False
DisplayControl:	Text Box
GUID:	{guid {CB360F25-14F4-42E3-AAB3-1E6624AAAA77}}
IMEMode:	0
IMESentenceMode:	3
OrdinalPosition:	3
Required:	True
SourceField:	Form Tutor
SourceTable:	Forms
TextAlign:	General
UnicodeCompression:	True
ValidationRule:	Like "mr *" Or Like "mrs *" Or Like "miss *" Or Like "ms *"
ValidationText:	Form Tutors must begin with Mr, Ms, Mrs or Miss.

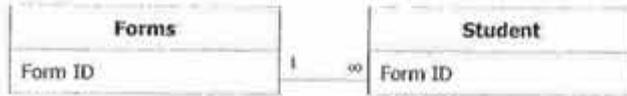
- field must be completed

validation for - Start of name

Text shown if validation not met.

Relationships

**FormsStudent**



Attributes: Enforced  
RelationshipType: One-To-Many

One-to-many relationship between 'Form Groups' and 'Students' table.

**Table Indexes**

Name	Number of Fields
PrimaryKey	1
Clustered:	False
DistinctCount:	12
Foreign:	False
IgnoreNulls:	False
Name:	PrimaryKey
Primary:	True
Required:	True
Unique:	True
Fields:	
Form ID	Ascending

'Form ID' is primary key in ascending order.

- Following pages show 'loans' table

**Properties**

DateCreated:	05/07/2009 22:30:28	DefaultView:	2
DisplayViewsOnSharePointSit	1	FilterOnLoad:	False
GUID:	{guid {9D533AC5-818C-48BF-933E-9A05A680682B}}	HideNewField:	False
LastUpdated:	18/03/2010 17:59:08	NameMap:	Long binary data
OrderByOn:	False	OrderByOnLoad:	True
Orientation:	Left-to-Right	RecordCount:	17
TotalsRow:	False	Updatable:	True

**Columns**

Name	Type	Size
<b>Loan ID</b>	Long Integer	4
AggregateType:	-1	
AllowZeroLength:	False	
AppendOnly:	False	
Attributes:	Fixed Size, Auto-Increment	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	1	
ColumnWidth:	Default	
DataUpdatable:	False	
GUID:	{guid {31DC39F0-8659-4C94-933B-56F9A6CB8860}}	
OrdinalPosition:	0	
Required:	False	
SourceField:	Loan ID	
SourceTable:	Loans	
TextAlign:	General	

- 'Loan ID' is the Primary Key

Name	Type	Size
<b>Date of Loan</b>	Date/Time	8
AggregateType:	-1	
AllowZeroLength:	False	
AppendOnly:	False	
Attributes:	Fixed Size	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	Default	
ColumnWidth:	1575	
DataUpdatable:	False	
DefaultValue:	=Now()	
Format:	Short Date	
GUID:	{guid {15261714-701C-441B-BC4A-51B7F38E7070}}	
IMEMode:	0	
IMESentenceMode:	3	
InputMask:	##/##/####	
OrdinalPosition:	1	
Required:	True	
ShowDatePicker:	For dates	
SourceField:	Date of Loan	
SourceTable:	Loans	
TextAlign:	General	

- Field cannot be empty

- Field as default shows today's date

- Forces date/time to ignore 'time' section if date picker used. must be completed Date picker can be used.

Table: Loans

Page: 9  
8

Date Due Back

Date/Time

AggregateType:	-1
AllowZeroLength:	False - Cannot be left empty
AppendOnly:	False
Attributes:	Fixed Size
CollatingOrder:	General
ColumnHidden:	False
ColumnOrder:	Default
ColumnWidth:	1740
DataUpdatable:	False
DefaultValue:	=Now()+7 - AS default, 1 week after today is shown.
Format:	Short Date
GUID:	{guid {D3DBBD5D-D0FA-430D-8603-03DC3C650C84}}
IMEMode:	0
IMESentenceMode:	3 - Forces date picker to ignore time stamp.
InputMask:	##/##/#### - field must be completed.
OrdinalPosition:	2 - Date picker shown.
Required:	True
ShowDatePicker:	For dates
SourceField:	Date Due Back
SourceTable:	Loans
TextAlign:	General
ValidationRule:	>Now()+7 - date must be more than 7 days after today
ValidationText:	The date due back must be after the date of the loan.

Returned?

Yes/No

AggregateType:	-1
AllowZeroLength:	False - field cannot be empty. Text shown if validation not met.
AppendOnly:	False
Attributes:	Fixed Size
CollatingOrder:	General
ColumnHidden:	False
ColumnOrder:	Default
ColumnWidth:	Default
DataUpdatable:	False
DisplayControl:	106
Format:	Yes/No
GUID:	{guid {CE7D6CA1-FA66-4E5D-A5DA-00DBEE70622}}
OrdinalPosition:	3
Required:	False - Does not have to be required as left blank check box until book returned.
SourceField:	Returned?
SourceTable:	Loans
TextAlign:	General

Book ID

Long Integer

4

AggregateType:	-1
AllowZeroLength:	False - Cannot be empty.
AppendOnly:	False
Attributes:	Fixed Size
CollatingOrder:	General
ColumnHidden:	False
ColumnOrder:	Default
ColumnWidth:	Default
DataUpdatable:	False
DecimalPlaces:	Auto
DisplayControl:	Text Box

Foreign key.

GUID: {guid {B2E00C5B-4EF7-4CE9-B0F8-0EF792E576F1}}  
 OrdinalPosition: 4  
 Required: True  
 SourceField: Book ID  
 SourceTable: Loans  
 TextAlign: General

*- must be input.*

**Student ID**

Long Integer

4

AggregateType: -1  
 AllowZeroLength: False  
 AppendOnly: False  
 Attributes: Fixed Size  
 CollatingOrder: General  
 ColumnHidden: False  
 ColumnOrder: Default  
 ColumnWidth: Default  
 DataUpdatable: False  
 DecimalPlaces: Auto  
 DisplayControl: Text Box  
 GUID: {guid {1656A7A3-2B3A-4FD0-942D-5A6E74532ECF}}  
 OrdinalPosition: 5  
 Required: True  
 SourceField: Student ID  
 SourceTable: Loans  
 TextAlign: General

*Cannot be left blank*

*- Required to input data.*

*foreign key*

**Relationships**

**BooksLoans**



Attributes: Enforced  
 RelationshipType: One-To-Many

*One-to-many relationships between books and loans, and students and loans.*

**StudentLoans**



Attributes: Enforced  
 RelationshipType: One-To-Many

**Table Indexes**

Name	Number of Fields
Book ID	1
Clustered:	False
DistinctCount:	13

Table: Loans

Page: 11

Foreign:	False
IgnoreNulls:	False
Name:	Book ID
Primary:	False
Required:	False
Unique:	False
Fields:	
Book ID	Ascending
<b>BooksLoans</b>	<b>1</b>
Clustered:	False
DistinctCount:	13
Foreign:	True
IgnoreNulls:	False
Name:	BooksLoans
Primary:	False
Required:	False
Unique:	False
Fields:	
Book ID	Ascending
<b>PrimaryKey</b>	<b>1</b>
Clustered:	False
DistinctCount:	17
Foreign:	False
IgnoreNulls:	False
Name:	PrimaryKey
Primary:	True
Required:	True
Unique:	True
Fields:	
Loan ID	Ascending
<b>Student ID</b>	<b>1</b>
Clustered:	False
DistinctCount:	16
Foreign:	False
IgnoreNulls:	False
Name:	Student ID
Primary:	False
Required:	False
Unique:	False
Fields:	
Student ID	Ascending
<b>StudentLoans</b>	<b>1</b>
Clustered:	False
DistinctCount:	16
Foreign:	True
IgnoreNulls:	False
Name:	StudentLoans
Primary:	False
Required:	False
Unique:	False
Fields:	
Student ID	Ascending

*- Following pages show 'Students' table*

**Properties**

DateCreated:	05/07/2009 22:21:30	DefaultView:	2
DisplayViewsOnSharePointSit 1		FilterOnLoad:	False
GUID:	{guid {DC714FC8-5AED-4B17-822C-D40133D5A881}}	HideNewField:	True
LastUpdated:	18/03/2010 17:58:45	NameMap:	Long binary data
OrderByOn:	False	OrderByOnLoad:	True
Orientation:	Left-to-Right	RecordCount:	31
TotalsRow:	False	Updatable:	True

**Columns**

Name	Type	Size
Student ID	Long Integer	4
AggregateType:	-1	
AllowZeroLength:	False	
AppendOnly:	False	
Attributes:	Fixed Size, Auto-Increment	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	1	
ColumnWidth:	Default	
DataUpdatable:	False	
Format:	General Number	
GUID:	{guid {8908F656-106D-4387-AB1E-D37EAAE242A4}}	
OrdinalPosition:	0	
Required:	False	
SourceField:	Student ID	
SourceTable:	Student	
TextAlign:	General	

*'Student ID' is primary key.*

First Name	Text	20
AggregateType:	-1	
AllowZeroLength:	True	
AppendOnly:	False	
Attributes:	Variable Length	
CollatingOrder:	General	
ColumnHidden:	False	
ColumnOrder:	Default	
ColumnWidth:	Default	
DataUpdatable:	False	
DisplayControl:	Text Box	
GUID:	{guid {C9FA8300-2157-4F5F-872E-8D6E7B499AED}}	
IMEMode:	0	
IMESentenceMode:	3	
OrdinalPosition:	1	
Required:	True	
SourceField:	First Name	
SourceTable:	Student	
TextAlign:	General	
UnicodeCompression:	True	
ValidationRule:	Not Like "[a-z]*"	
ValidationText:	First Name must contain letters and no numbers.	

*- Field must be completed.*

*Format is 'text' so will allow letters + numbers. This validation rejects any numbers.*

*Error shown if numbers are entered.*

Table: Student

Page: 13  
30

Last Name

Text

AggregateType: -1  
 AllowZeroLength: True  
 AppendOnly: False  
 Attributes: Variable Length  
 CollatingOrder: General  
 ColumnHidden: False  
 ColumnOrder: Default  
 ColumnWidth: Default  
 DataUpdatable: False  
 DisplayControl: Text Box  
 GUID: {guid {8BC1BF04-F354-4A63-9A9B-2C310DFEBC16}}  
 IMEMode: 0  
 IMESentenceMode: 3  
 OrdinalPosition: 2  
 Required: True  
 SourceField: Last Name  
 SourceTable: Student  
 TextAlign: General  
 UnicodeCompression: True  
 ValidationRule: Not Like "[a-z]\*"  
 ValidationText: Last Name must contain letters and no numbers.

- must be completed.

- Rejects numbers from being input

- error shown if numbers are input

Additional Notes

Text

200

AggregateType: -1  
 AllowZeroLength: True  
 AppendOnly: False  
 Attributes: Variable Length  
 CollatingOrder: General  
 ColumnHidden: False  
 ColumnOrder: Default  
 ColumnWidth: 3180  
 DataUpdatable: False  
 DisplayControl: Text Box  
 GUID: {guid {EC821BE8-6792-44D1-BF81-EA6382DB79D9}}  
 IMEMode: 0  
 IMESentenceMode: 3  
 OrdinalPosition: 3  
 Required: False  
 SourceField: Additional Notes  
 SourceTable: Student  
 TextAlign: General  
 UnicodeCompression: True

Additional notes not required and can be left empty - these are optional.

Form ID

Long Integer

4

AggregateType: -1  
 AllowZeroLength: False  
 AppendOnly: False  
 Attributes: Fixed Size  
 CollatingOrder: General  
 ColumnHidden: False  
 ColumnOrder: Default  
 ColumnWidth: Default  
 DataUpdatable: False  
 DecimalPlaces: Auto  
 DisplayControl: Text Box

- field cannot be empty

Foreign key

Table: Student

Page: 14

GUID: {guid (FDB1395C-6829-4D6E-9995-144E833B4AC9)}  
 OrdinalPosition: 4  
 Required: True  
 SourceField: Form ID  
 SourceTable: Student  
 TextAlign: General

- Field is required

**Relationships**

**FormsStudent**



Attributes: Enforced  
 RelationshipType: One-To-Many

One-to-many relationship for Forms table to Student table.

**StudentLoans**



Attributes: Enforced  
 RelationshipType: One-To-Many

one-to-many relationship for Student table to loans table.

**Table Indexes**

Name	Number of Fields
Form ID	1
Clustered:	False
DistinctCount:	12
Foreign:	False
IgnoreNulls:	False
Name:	Form ID
Primary:	False
Required:	False
Unique:	False
Fields:	
Form ID	Ascending
FormsStudent	1
Clustered:	False
DistinctCount:	12
Foreign:	True
IgnoreNulls:	False
Name:	FormsStudent
Primary:	False
Required:	False
Unique:	False

Table: Student

Page: 15

Fields:	
Form ID	Ascending
PrimaryKey	1
Clustered:	False
DistinctCount:	31
Foreign:	False
IgnoreNulls:	False
Name:	PrimaryKey
Primary:	True
Required:	True
Unique:	True
Fields:	
Student ID	Ascending

Test Number	Test	Expected Result
<b>Miscellaneous Tests</b>		
1	Opening Database	Database Opens correctly when file is opened.
2	Closing Database	Database closes when exit button is selected.
3	About button	When selected the button opens the About menu
4	Reports button	When selected the Reports Menu opens
5	About Menu	About Menu displays correctly and is error-free
6	Table relationships	Referential integrity is working - A student cannot be deleted if they have a current loan.
7	Reports Menu	Reports menu displays and is error-free.
<b>Student Table</b>		
8	Student ID	Student ID is working correctly
9	First Name	First name displays Data correctly
10	Last Name	Last name displays data correctly
11	Additional Notes	Additional Notes displays correctly.
12	Form ID	Form ID displays Correctly.
13	First Name validation rule	Validation rule for First name works (No numbers)
14	Last Name validation rule	Validation rule for Last name works (No numbers)
15	Auto number	Auto number for Student ID is working
16	All fields complete validation.	All fields must be filled to enter new record.
<b>Books Table</b>		
17	Book ID	Book ID displays correctly
18	ISBN Number	ISBN number is displayed
19	Book Name	Book name can be entered
20	Author Name	Author can be entered
21	Publisher	Publisher displays correctly
22	Genre	Genre can be entered and displays correctly
23	Price	Price can be input
24	Auto number	Auto number for Book ID is working correctly.
25	Input Mask	Input Mask for ISBN number is working correctly.
26	Book price validation	Book Price validation rule works (must be greater than 0)
27	Book price field type	Book price automatically set to currency is pounds.
28	All fields complete validation.	All fields must be completed to enter new book.
<b>Forms Table</b>		

	Form ID	Form ID displays correctly.
30	Year Group	Year group can be entered and displays.
31	Form Room	Form Room can be input
32	Form Tutor	Form Tutor name can be input.
33	Auto number	Auto number for Form ID working correctly.
34	Year group validation rule	Year group validation rule working (must be between year 7 and 13)
35	Form Room input mask	Form Room Input mask working which stops incorrect format Room numbers being entered.
36	Form room validation rule	Form room Validation rule working (Must begin with TA, TB or TC)
37	Form tutor validation rule	Form Tutor validation rule working (Begin with Mr, Mrs, Miss or Ms"
38	All fields complete validation.	All fields must be completed to enter new form group
<b>Loans Table</b>		
39	Loan ID	Loan ID displaying correctly and is automatically numbered.
40	Date of Loan	Date of loan displaying with calendar
41	Date due back	Date due back displaying with calendar
42	Returned Check Box	Returned check box working correctly.
43	Book ID	Book ID can be entered
44	Student ID	Student ID can be entered
45	Date due back validation rule	Validation rule stops the date due back being before the date of the loan
46	All fields complete validation.	All fields should be completed before a record can be added.
<b>Student Loans Form</b>		
47	Student Loans Form opens	Student Loans Form opens when the button is selected.
48	Data displaying correctly.	Data Displays in Student Loan form correctly - all data can easily be seen.
49	Time and date	The time and date should correctly display the system time and date in the bottom left part of the form.
50	Add New Loan	When the button is selected, A new loan should be added to the database.
51	Delete Loan	The loan should be deleted from the database.
52	Save loan	Changes that are made are saved correctly.
53	Print loan	The currently displayed loan should be sent to the printer.
54	Navigation options	Navigation options should allow navigation to the first, last, previous and next records as well as a search facility.

Books Form		
55	Books form opens	The Books Form opens correctly from button on the main menu
56	Data displaying correctly.	Data Displays in Books form correctly
57	Save Book	The button should save any changes to a book record.
58	Add New Book	This should add a new record to the Books table.
59	Delete Book	This should remove a Book from the table.
60	Print Book	This button should send the currently displayed Book record to the printer.
61	Time and date	The time and date should correctly display the system time and date in the bottom left part of the form.
62	Navigation options	Navigation options should allow navigation to the first, last, previous and next records as well as a search facility.
Students Form		
63	Students Form opens	The form should open when the button is selected from the main menu.
64	Data is displaying correctly	All data within the form should be visible and display correctly.
65	Print Button	The print button should print the current record when selected.
66	Delete Button	This button should delete any current students from the database
67	Add students Button	A new record should be created when this button is selected.
68	Save Button	The save button should save changes to the current record when selected.
69	Time and date	The time and date should display correctly on the form.
70	Navigation options	Navigation options should allow navigation to the first, last, previous and next records as well as a search facility.
Form groups Form		
71	Data displaying correctly.	All data should display correctly and be clearly visible within the form
72	Save button	The save button should save any changes that have been made
73	Print Button	The current record should be printed when this button is selected.
74	Delete Button	This button should delete the record currently shown within the form.
75	Add new Form Group button	This button should add a new form group to the database.

74	Delete Button	This button should delete the record currently shown within the form.
75	Add new Form Group button	This button should add a new form group to the database.
76	Navigation options	These options should allow you to search for a Form ID or navigate to the first, last, previous or next record.
77	Date and time	The time and date should display correctly on the form.
<b>Queries</b>		
78	Query 1	All Books by author works correctly for any given author. The test Value will be the Author "David Rayner" and the results should be Book ID's 15, 23, 28, 29, 30, 31, 32, 33, 34, 39 and 41
79	Query 2	All books of a particular genre works for any given genre. The Test Genre will be ICT. Expected results are Book ID's 13, 16, 19, 35 and 42
80	Query 3	All books out to a student works correctly for any student ID entered. The test Student ID is 17 which should produce two books that are on loan, ID's 16 and 13
81	Query 4	All overdue books for a particular form displays correct results for any Form ID. The test Form ID is 15 which should produce overdue loans for Book ID's 13, 16 and 17
82	Query 5	All students that have an overdue book by 2 weeks displays correct results. When this query is run the results should be Book ID's 16,17,20,21 and 24
83	Query 6	Books of a genre due back between 2 days works correctly by asking for the genre and two dates. The test dates will be between 15/7/2009 and 10/8/2009 with the Genre of ICT. The results are expected to be 13, 16 and 19.
84	Query 7	All books due back on a particular date displays correct results for any given date. Expected results are book ID's 15 and 35 for date 17/03/2010
85	Query 8	All books due back today displays correct results for that particular day. The test date of 18/03/2010 when this is run should produce loan ID 53.

86	Query 9	All books of a particular genre by an author displays correct results. Test genre is ICT and Test Author os "Julian Mott". This should display results for Book ID's 24,25,27, 38,40 and 42
87	Query 10	Students found by first and last name when entered into paramerter query displays correct results. I am going to search for student "\ to ensure that results are displayed correctly.
<b>Reports</b>		
88	Report 1	All books by author displays same reuslts as query with designed graphics and layout.
89	Report 2	All books of a particular genre displays the same results as the query with the layout and graphics I have designed
90	Report 3	All books loaned to a particular student displays same results as query with correct design layout
91	Report 4	All overdue books for a particular form displays the same reuslts as query on desinged letter layout.
92	Report 5	All students that have an overdue book by 2 weeks displays the same results as query on my designed report.
93	Report 6	Books of a particular genre between two dates should display same results as the tested query with results displayed in report format
94	Report 7	All books back on a particular date produces same reuslts as query with a user-friendly layout.
95	Report 8	All books due back today produces the same results as the query showed but on a more user-friendly graphically designed report.
96	Report 9	All books of a particular genre by author should also produce the same results as the query again printing in the designed layout of the report so that it is easier to read.
97	Report 10	Students found by first and last name should produce the same query results where results are shown on a more user-friendly display.

Test Number	Test	Expected Result	Actual Result	Corrective action
<b>Miscellaneous Tests</b>				
1	Opening Database	Database Opens correctly when file is opened.	The database opens correctly with the splash screen displaying for a few seconds.	None required.
2	Closing Database	Database closes when exit button is selected.	Once the exit button is selected, a message appears asking for confirmation before closing the database.	None required.
3	About button	When selected the button opens the About menu	Once this button is selected the about menu is displayed above the main menu.	None required.
4	Reports button	When selected the Reports Menu opens	Once the reports button has been clicked by the user, the reports menu is opened above the main menu.	None required.
5	About Menu	About Menu displays correctly and is error-free	When displayed, the about menu is clearly visible and text is error-free.	None required.
6	Table relationships	Referential integrity is working - A student cannot be deleted if they have a current loan.	A student could not be deleted because he currently has a loan, therefore this is working correctly.	None required.
7	Reports Menu	Reports menu displays and is error-free.	The reports main clearly displays all available reports. Text used on the menu is accurate and error-free.	None required.
<b>Student Table</b>				
8	Student ID	Student ID is working correctly	The value for Student ID can be clearly seen within the table.	None required.
9	First Name	First name displays Data correctly	The first name field is displayed clearly within the table.	None required.
10	Last Name	Last name displays data correctly	The last name field is displayed clearly within the table.	None required.
11	Additional Notes	Additional Notes displays correctly.	The additional notes section allows entry of any text for any notes that may wish to be made about students.	None required.
12	Form ID	Form ID displays Correctly.	The form ID field is also displaying the form ID value correctly.	None required.

13	First Name validation rule	Validation rule for First name works (No numbers)	The validation rule is working correctly when a number is entered into the field. This validation rule stops incorrect data being entered into the table by showing the error "First name must contain letters and no numbers".	None required.
14	Last Name validation rule	Validation rule for Last name works (No numbers)	This validation rule is also working, meaning number cannot be entered for the last name field. See supporting documentation for screenshot of error message shown.	None required.
15	Auto number	Auto number for Student ID is working	When a new record is entered into the table, it is automatically assigned a new unique student ID number.	None required.
16	All fields complete validation.	All fields must be filled to enter new record.	An error message is produced should a required field not be completed. All fields except for "additional notes" are required.	None required.
<b>Books Table</b>				
17	Book ID	Book ID displays correctly	It can be seen from my supporting screenshots that the Book ID number is displaying correctly.	None required.
18	ISBN Number	ISBN number is displayed	The ISBN number is displayed correctly with numbers and no letters allowed to be input	None required.
19	Book Name	Book name can be entered	The Book Name field allows data input and the data can be clearly seen.	None required.
20	Author Name	Author can be entered	The Author name can also be entered with data being clearly visible within the table.	None required.
21	Publisher	Publisher displays correctly	The publisher name can also be input with the data being displayed correctly.	None required.
22	Genre	Genre can be entered and displays correctly	The Genre can also be input into the database correctly.	None required.

23	Price	Price can be input	The price can be input into the table with numbers and no letters being input..	None required.
24	Auto number	Auto number for Book ID is working correctly.	The Book ID field is automatically numbered correctly with each record being assigned a unique number.	None required.
25	Input Mask	Input Mask for ISBN number is working correctly.	The Input mask for the ISBN number correctly formats the input of data into the correct length and layout.	None required.
26	Book price validation	Book Price validation rule works (must be greater than 0)	The book price validation works correctly by ensuring that the field cannot be left blank or £0.00. This validation rule therefore accepts correct data and rejects incorrect data (£0.00) from being entered into the system.	None required.
27	Book price field type	Book price automatically set to currency is pounds.	The book price field type automatically sets the field format to a currency in pounds.	None required.
28	All fields complete validation.	All fields must be completed to enter new book.	Only the Author name field was required, meaning the validation was not working for all other fields. After corrective action had been taken, all fields required information in order to save a record.	I had to go back into design view where I found that although the "field required" was set to "yes", the "allow zero length" was set to "yes", meaning the field could be left blank therefore I changed this to "no"
<b>Forms Table</b>				
29	Form ID	Form ID displays correctly.	The form ID that is generated can be clearly seen within the table.	None required.
30	Year Group	Year group can be entered and displays.	The year group can be entered into the table and is clearly displayed	None required.

31	Form Room	Form Room can be input	The form room is also clearly displayed within the table once data has been input into the field.	None required.
32	Form Tutor	Form Tutor name can be input.	The form Tutors name is also easily viewable within the table.	None required.
33	Auto number	Auto number for Form ID working correctly.	The Form ID has a new number assigned for every record which is unique meaning it will never be repeated.	None required.
34	Year group validation rule	Year group validation rule working (must be between year 7 and 13)	After entering "25" the error message was displayed therefore meaning the validation rule is working correctly.	None required.
35	Form Room input mask	Form Room input mask working which stops incorrect format Room numbers being entered.	The input mask is working correctly as the form room can only be entered in the format of Letter, Letter, Number, Number	None required.
36	Form room validation rule	Form room Validation rule working (Must begin with TA, TB or TC)	The form room validation rule is working correctly, only allowing the data to start with TA, TB or TC.	None required.
37	Form tutor validation rule	Form Tutor validation rule working (Begin with Mr, Mrs, Miss or Ms"	This validation rule works correctly stopping any incorrect starts to form tutors name. It will also allow correct data entries to pass. This has been tested by entering correct names starting in "Mr" and incorrect data as "Mt" should a mistake be made.	None required.
38	All fields complete validation.	All fields must be completed to enter new form group	All fields within the table require data to be entered before the record can be saved therefore this is working correctly.	None required.
<b>Loans Table</b>				
39	Loan ID	Loan ID displaying correctly and is automatically numbered.	The loan ID is displaying correctly with the auto number providing a unique ID number for each loan made.	None required.

40	Date of Loan	Date of loan displaying with calendar	The date of the loan is also displaying correctly with the date picker appearing when selected.	None required.
41	Date due back	Date due back displaying with calendar	Once the date due back is selected the date can be input in the correct format with the calendar shown if selected.	None required.
42	Returned Check Box	Returned check box working correctly.	The returned checkbox is clearly visible and allows the checkbox to be both selected and de-selected.	None required.
43	Book ID	Book ID can be entered	The Book ID can be input and is clearly shown within the table. Duplicate books can also be entered within the table as they will go on loan multiple times.	None required.
44	Student ID	Student ID can be entered	The student ID can also be entered and is clearly displayed within the table. Once again, Duplicate entries can also be made for the Same student should they wish to borrow more than one book.	None required.
45	Date due back validation rule	Validation rule stops the date due back being before the date of the loan	The date due back has to be after the date of the loan otherwise an error message is produced. Both a date before and after the loan have been tried. The date before the loan was rejected and after was accepted.	None required.
46	All fields complete validation.	All fields should be completed before a record can be added.	This validation is working correctly ensuring that all fields are completed before a new record can be added to the database.	None required.
<b>Student Loans Form</b>				
47	Student Loans Form opens	Student Loans Form opens when the button is selected.	The loan form opens correctly once the button is selected.	None required.

	Data displaying correctly.	Data Displays in Student Loan form correctly - all data can easily be seen.	The data shown is fairly correct however one section of the form is not updating and the alignment of one fields data is incorrect.	I have gone back into design view and re-added the "form groups" as a new subform, hiding the relevant data. I have also edited the properties of the sub-forms and re-aligned text which was out of line.
48	Time and date	The time and date should correctly display the system time and date in the bottom left part of the form.	The time does not update unless the page is refreshed however the time and date are otherwise displaying correctly.	None required - See documentation as a macro to update the page may cause more major problems with the database.
49	Add New Loan	When the button is selected, A new loan should be added to the database.	The Add new loan button worked correctly allowing me to add a new loan to the database.	None required.
50	Delete Loan	The loan should be deleted from the database.	The delete loan button worked correctly, deleting the loan I had previously added to the database, producing a warning message first for confirmation.	None required.
51	Save loan	Changes that are made are saved correctly.	This button also worked correctly with the new loan being saved onto the database.	None required.
52	Print loan	The currently displayed loan should be sent to the printer.	The print loan button sent the current record to the printer with a print dialogue asking the user things such as number of copies and which printer to send it to.	None required.
53				

76	Navigation options	These options should allow you to search for a Form ID or navigate to the first, last, previous or next record.	The navigation options within the form also work correctly, effectively allowing me to navigate to the first, last, previous and next records as well as search for a specific Form ID. See documentation for Test 54 as these buttons were duplicated for this form.	None required.
77	Date and time	The time and date should display correctly on the form.	Finally the date and time are also displaying correctly within the form with the exception of the time not updating. See text 49 for documentation as this code was duplicated.	The error with the time is noted however would cause more problems if fixed.
<b>Queries</b>				
78	Query 1	All Books by author works correctly for any given author. The test Value will be the Author "David Rayner" and the results should be Book ID's 15, 23, 28, 29, 30, 31, 32, 33, 34, 39 and 41	The correct results were displayed within the query. See printed query for documentation of this.	None Required.
79	Query 2	All books of a particular genre works for any given genre. The Test Genre will be ICT. Expected results are Book ID's 13, 16, 19, 35 and 42	Results displayed were correct however results were displayed multiple times within the database.	I have changed settings within the query so that results will only be displayed once.
80	Query 3	All books out to a student works correctly for any student ID entered. The test Student ID is 17 which should produce two books that are on loan, ID's 16 and 13	The correct results are displayed however unwanted fields are displayed named "student id" as the user already knows what this is and "Expr1000" which also shows the Student ID.	Two instances of "Student ID" were found. One was removed and the other was hidden as it is required to run the query.

81	Query 4	All overdue books for a particular form displays correct results for any Form ID. The test Form ID is 15 which should produce overdue loans for Book ID's 13, 16 and 17	The correct results were displayed from this query.	None Required.
82	Query 5	All students that have an overdue book by 2 weeks displays correct results. When this query is run the results should be Book ID's 16,17,20,21 and 24	The correct results were displayed when this query was run.	None Required.
83	Query 6	Books of a genre due back between 2 days works correctly by asking for the genre and two dates. The test dates will be between 15/7/2009 and 10/8/2009 with the Genre of ICT. The results are expected to be 13, 16 and 19.	The results are displaying correctly however date input to the parameter was not very user friendly. It had to be manually entered as dd/mm/yyyy which is hard to remember.	Input masks cannot be specified on the pop-up boxes from parameter queries, therefore I have simply changed the text to inform the user of the format it must be changed to.
84	Query 7	All books due back on a particular date displays correct results for any given date. Expected results are book ID's 15 and 35 for date 17/03/2010	The query results table was blank as now relevant results were found.	The "loans" table needed an input mask so that the system time was not taken into account.
85	Query 8	All books due back today displays correct results for that particular day. The test date of 18/03/2010 when this is run should produce loan ID 53.	This query was also not working correctly as the system was not finding "today's" date correctly.	The Criteria for the query needed changing from "Now" to "date" so that the time was not used.
86	Query 9	All books of a particular genre by an author displays correct results. Test genre is ICT and Test Author is "Julian Mott". This should display results for Book ID's 24,25,27, 38,40 and 42	This query is running correctly with all expected results displayed once the query is run.	None required

	Query 10	Students found by first and last name when entered into parameter query displays correct results. I am going to search for student "Oliver Harris" to ensure that results are displayed correctly.	Results were displayed for the names matching "Oliver" and "Harris" which displayed two results, these were "Oliver Harris" and "Oliver Staley". This query also produces both results should the query be run with only the first name "Oliver". It also works with only the last name as "Harris", producing the one student ID that matches.	None required
<b>Reports</b>				
88	Report 1	All books by author displays same results as query with designed graphics and layout.	This report displays the same results as its associated query and displays them in an easy to reads format.	None required
89	Report 2	All books of a particular genre displays the same results as the query with the layout and graphics I have designed	The report also displays results clearly in the designed layout.	None required
90	Report 3	All books loaned to a particular student displays same results as query with correct design layout	The report operates correctly with all information automatically generated from the query	None required
91	Report 4	All overdue books for a particular form displays the same results as query on designed letter layout.	The report contents are displayed however the same problem is occurring here as on the data input forms as the date is shown as "#####"	The field for the date has been increased in size in order to allow more room, therefore allowing the date to display correctly. See Test 49 for similar documentation of how this is resolved.
92	Report 5	All students that have an overdue book by 2 weeks displays the same results as query on my designed report.	This report automatically generated the information from the query displaying it in an easy-to-read way.	None required

93	Report 6	Books of a particular genre between two dates should display same results as the tested query with results displayed in report format	This report is also working correctly displaying data in a user-friendly format once the data has been entered into the query.	None required
94	Report 7	All books back on a particular date produces same results as query with a user-friendly layout.	The query is displayed correctly however the green banding put across the page to separate documents shows white text boxes which looks unprofessional.	The text boxes were set to be transparent so that the green colour could be seen.
95	Report 8	All books due back today produces the same results as the query showed but on a more user-friendly graphically designed report.	The report is displaying the same correct results as the query with no errors in the way that data is displayed.	None required
96	Report 9	All books of a particular genre by author should also produce the same results as the query again printing in the designed layout of the report so that it is easier to read.	This report clearly lists the books of a particular genre once the genre has been entered into the database.	None required
97	Report 10	Students found by first and last name should produce the same query results where results are shown on a more user-friendly display.	This final report is also displaying the students found by first or last name correctly, allowing the librarians clear identification of the students details.	None required

All books of a particular Genre - Fault from test 79.

17/03/2010

Book ID	ISBN Number	Book Name	Author	Publisher	Genre	Price
13	184-0-85602524-59-7	ICT GCSE Success	Sean O'Byrne	Letts	ICT	£3.50
13	184-0-85602524-59-7	ICT GCSE Success	Sean O'Byrne	Letts	ICT	£3.50
16	978-0-43546214-76-9	Applied ICT A2	Maggie Banks	Heinemann	ICT	£19.99
19	021-1-21556926-41-3	Running Visual Basic	Ross Nelson	Microsoft Press	ICT	£19.95
16	978-0-43546214-76-9	Applied ICT A2	Maggie Banks	Heinemann	ICT	£19.99
13	184-0-85602524-59-7	ICT GCSE Success	Sean O'Byrne	Letts	ICT	£3.50
16	978-0-43546214-76-9	Applied ICT A2	Maggie Banks	Heinemann	ICT	£19.99
35	184-0-85602524-59-7	ICT GCSE Success	Sean O'Byrne	Letts	ICT	£3.50
42	968-4-93748377-28-4	ICT for A Level 2nd Edition	Julian Mott	Hodder Murray	ICT	£29.99

This above query was produced from test 79 - It can be seen that the same book is appearing multiple times.

Expr1000	First Name	Last Name	Book ID	ISBN Number	Book Name	Author	Returned?	Student ID
17	Pollyanna	Ingley		16 978-0-43546214-76-9	Applied ICT A2	Maggie Banks	<input type="checkbox"/>	17
17	Pollyanna	Ingley		13 184-0-85602524-59-7	ICT GCSE Success	Sean O'Byrne	<input type="checkbox"/>	17

Results for test 80-'Student ID' is visible twice causing the 'Expr 1000' error.

Query: All books out to a particular student.

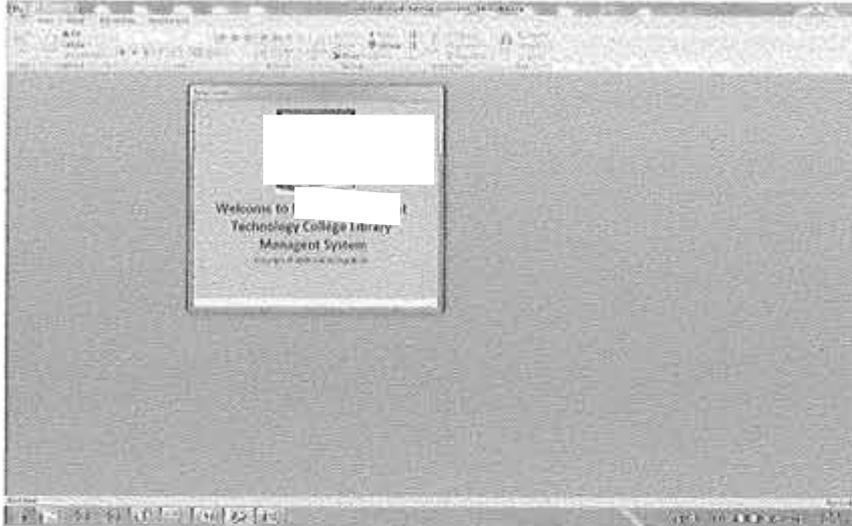
Genre	Price	Date of Loan	Date Due Back
ICT	£19.99	15/07/2009	06/08/2009
ICT	£3.50	13/04/2009	17/06/2009

Results for Test 80 Continued.

## Database Testing Documentation

I am now going to follow through my test plan, taking screenshot evidence of the testing that has been done. Some screenshots may be used for multiple tests if they show multiple elements of the database working.

### Test 1)



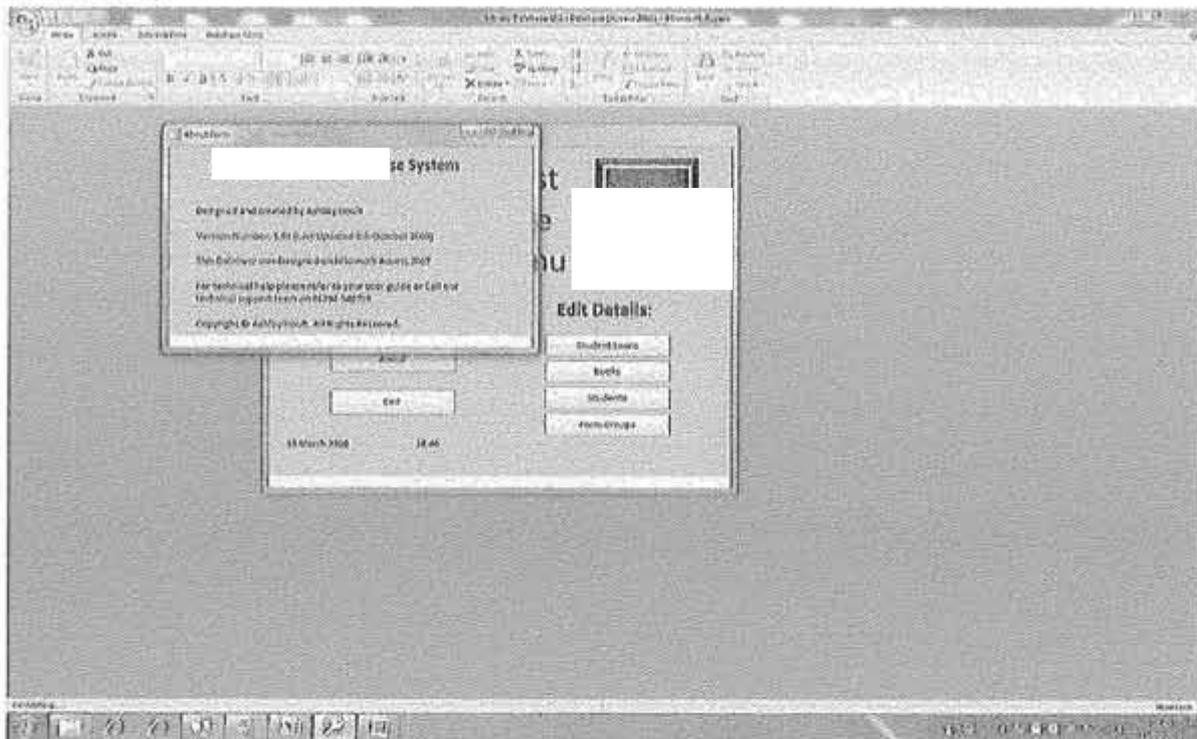
This above screenshot shows the screen that is displayed once the database is opened. This shows that the database can be opened successfully as the Splash Screen I have created displays for a few seconds before the main menu is shown.

### Test 2)



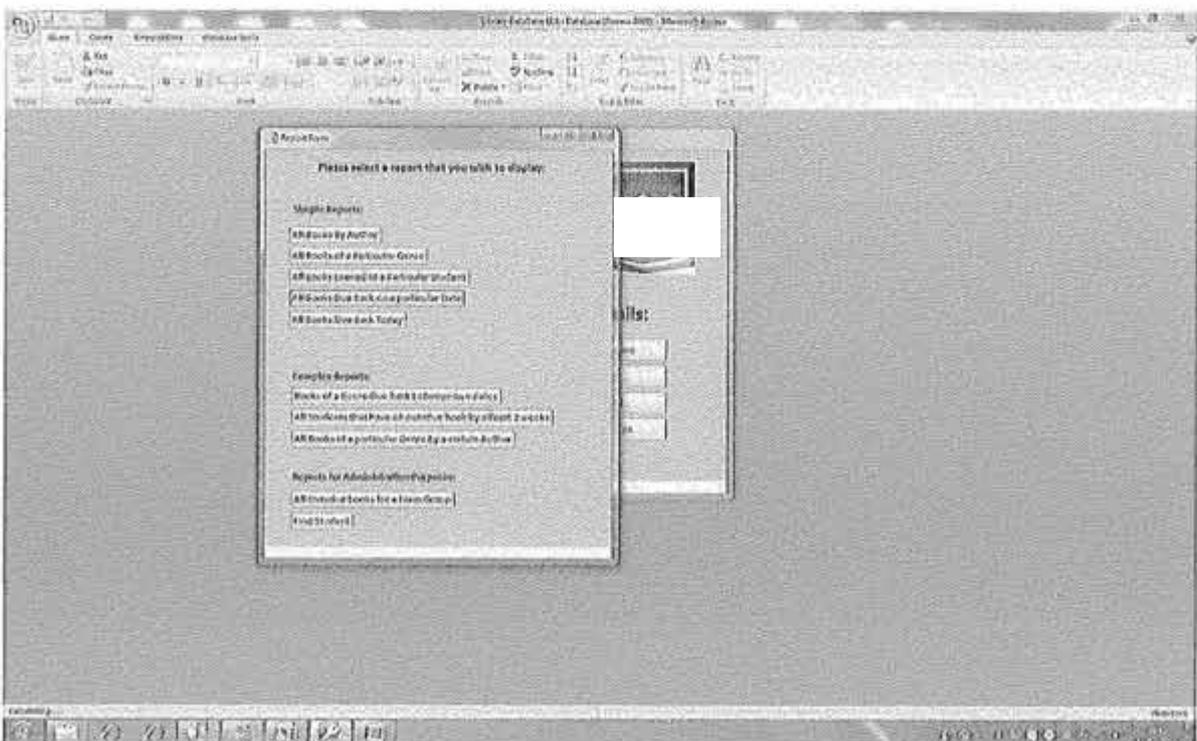
This screenshot shows test 2 working correctly. When the exit button is selected a message is loaded within a dialogue box asking for confirmation before the database is closed.

### Tests 3 & 5)



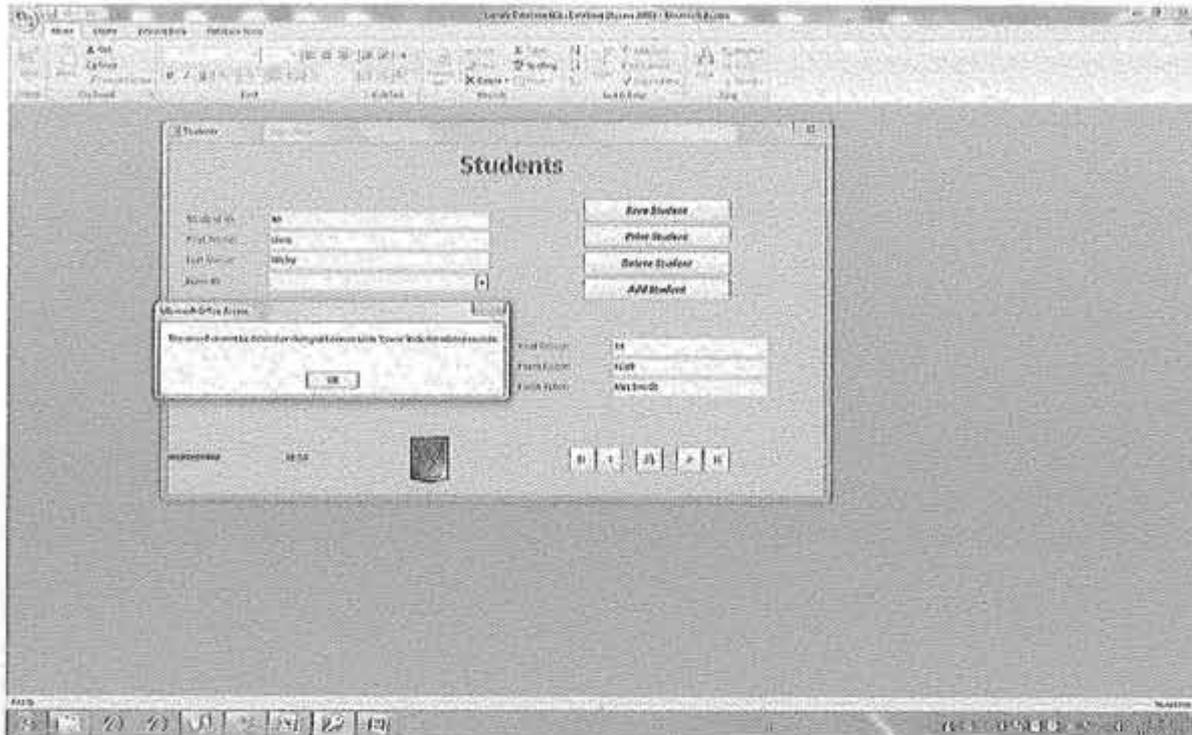
This screenshot shows both tests 3 and 5. After selecting the "about" button from the main menu, the "about" menu is shown. Test number 5 is to ensure that the about menu is displaying and is error-free. I can once again see from this screenshot that this is displaying correctly and after proof-reading, the text on here is also error-free.

### Tests 4 & 7)



This screenshot for Tests 4 and 7 ensures that the button for “reports” from the main menu is working correctly. After selecting this button, I can see that the menu is displayed correctly and that the “reports” menu is correct and error-free. This therefore shows that tests 4 and 7 have both been successful with no faults found.

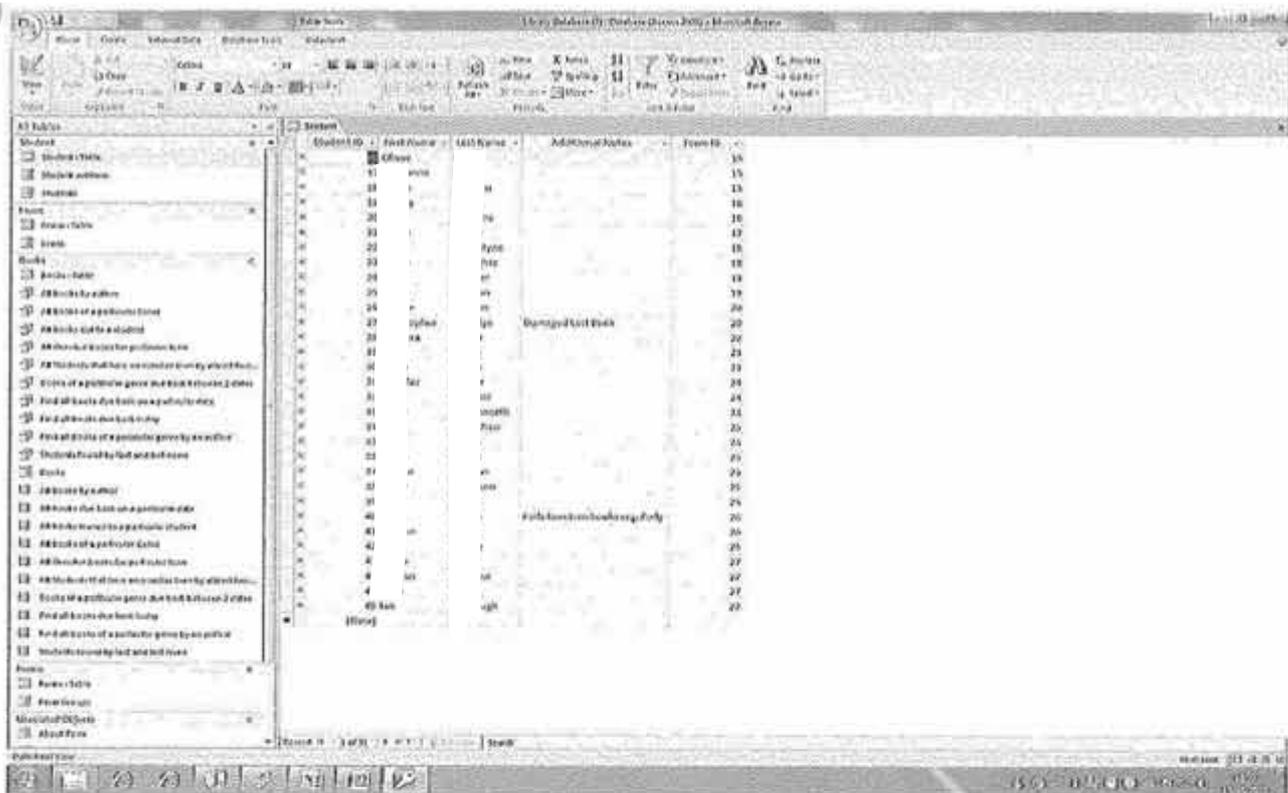
### Test 6)



This screenshot shows the testing of the referential integrity that I have used within the database. This means that a record cannot be deleted if there is another related record within the database. This was done by attempting to delete Student ID 45 who has already got a loan (ID 53). When attempting to delete the student, a message was produced saying that the record could not be deleted because the 'loans' table has a related record. This therefore shows that the referential integrity is working on my database.

### Tests 8 – 12 and 15)

The following screenshot for tests 8-12 shows the students Table that the database is run from. This table is hidden from the end user; however it must be tested to ensure data input into the table is working correctly.

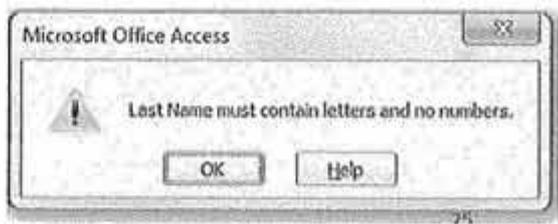


It can be clearly seen from this screenshot that the various fields within the “student” table are working correctly with the data that I have input to test the tables. Firstly I can see that the student ID fields are working correctly as the numbers have been automatically entered each time I have added a new student (Test 15). I can then also see that I have been able to input data in the first and last name fields with the data being displayed correctly. I have also used names like “grace” multiple times to ensure that people with the same or similar names can be entered into the database. Finally I have also included “additional notes” on two students and linked the ‘form id’ to each student, both of which again displayed correctly.

**Tests 13 and 14)**



This screenshot shows when I have entered a number into the “first name” field, a dialogue box is produced saying that the value cannot be accepted as you can use letters and no numbers. Similarly, this message is also produced when a number is entered into the “last name” field.



## Test 16)

Student ID	First Name	Last Name	Additional Notes	Form ID
16	r	na		15
17	ra	y		15
18	ra	on		15
19	ry			16
20	r	ms		16
21	r	f		17
22	r	lyne		18
23	r	lax		18
24	r	ur		19
25	r	an		19
26	ra	in		20
27	rophae	lge	Damaged last book	20
28	rcca	r		22
29	r	r		23
30	r	r		25
31	iler	r		25
32	ler			25
33	y	rumph		25
34	r	shaw		25
35	r	h		25
36	r	d		25
37	ne	lon		25
38	y	dson		25
39	r	h		25
40	r	th	fail to return books regularly	26
41	fon	n		26
42	r	ry		26
43	r	r		27
44	r	row		27
45	r	y		27
46	r	lough		27

This test has been completed to ensure that all fields within the table must be completed except for the "additional notes" field. This screenshot shows the error message produced when I have tried entering a new record without inputting a "form ID". This validation is therefore working in order to stop records being created with only partial data input.

## Tests 17-24)

Book ID	ISBN	Book Name	Author	Publisher	Genre	Price
1	0-01-010101-0	Microsoft Access	John O'Brien	Galle	IT	45.50
2	0-01-010101-1	Health & Social Care for HSC	Sylvia Meach	Wiley	Health & Social Care	35.50
3	0-01-010101-2	Using Mathematics in Health	David Kaye	Elsevier	Mathematics	49.95
4	0-01-010101-3	Applied KFFA	Maggie Books	Wiley	IT	118.50
5	0-01-010101-4	Applied KFFA	Ed May	Elsevier	Health	117.50
6	0-01-010101-5	Applied KFFA	David Kaye	Wiley	Business Studies	111.50
7	0-01-010101-6	Applied KFFA	Ross Nelson	Microsoft Press	IT	102.50
8	0-01-010101-7	Applied KFFA	Maggie Books	Wiley	IT	88.50
9	0-01-010101-8	Applied KFFA	Ross Nelson	Microsoft Press	IT	118.50
10	0-01-010101-9	Applied KFFA	David Kaye	Wiley	IT	130.00
11	0-01-010101-0	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
12	0-01-010101-1	Applied KFFA	David Kaye	Wiley	IT	130.00
13	0-01-010101-2	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
14	0-01-010101-3	Applied KFFA	David Kaye	Wiley	IT	130.00
15	0-01-010101-4	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
16	0-01-010101-5	Applied KFFA	David Kaye	Wiley	IT	130.00
17	0-01-010101-6	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
18	0-01-010101-7	Applied KFFA	David Kaye	Wiley	IT	130.00
19	0-01-010101-8	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
20	0-01-010101-9	Applied KFFA	David Kaye	Wiley	IT	130.00
21	0-01-010101-0	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
22	0-01-010101-1	Applied KFFA	David Kaye	Wiley	IT	130.00
23	0-01-010101-2	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
24	0-01-010101-3	Applied KFFA	David Kaye	Wiley	IT	130.00
25	0-01-010101-4	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
26	0-01-010101-5	Applied KFFA	David Kaye	Wiley	IT	130.00
27	0-01-010101-6	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
28	0-01-010101-7	Applied KFFA	David Kaye	Wiley	IT	130.00
29	0-01-010101-8	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
30	0-01-010101-9	Applied KFFA	David Kaye	Wiley	IT	130.00
31	0-01-010101-0	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
32	0-01-010101-1	Applied KFFA	David Kaye	Wiley	IT	130.00
33	0-01-010101-2	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
34	0-01-010101-3	Applied KFFA	David Kaye	Wiley	IT	130.00
35	0-01-010101-4	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
36	0-01-010101-5	Applied KFFA	David Kaye	Wiley	IT	130.00
37	0-01-010101-6	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
38	0-01-010101-7	Applied KFFA	David Kaye	Wiley	IT	130.00
39	0-01-010101-8	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
40	0-01-010101-9	Applied KFFA	David Kaye	Wiley	IT	130.00
41	0-01-010101-0	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
42	0-01-010101-1	Applied KFFA	David Kaye	Wiley	IT	130.00
43	0-01-010101-2	Applied KFFA	Ross Nelson	Microsoft Press	IT	130.00
44	0-01-010101-3	Applied KFFA	David Kaye	Wiley	IT	130.00

This screenshot shows the tests to ensure that data in all of the fields for the "Books" table can be entered and are displayed correctly. Firstly the "Book ID" field is working correctly with numbers being automatically added once a new record is created. These numbers are in a logical order and are unique meaning they will only ever appear once. The next field I have tested is the "ISBN" number field. This field is also working correctly with data from

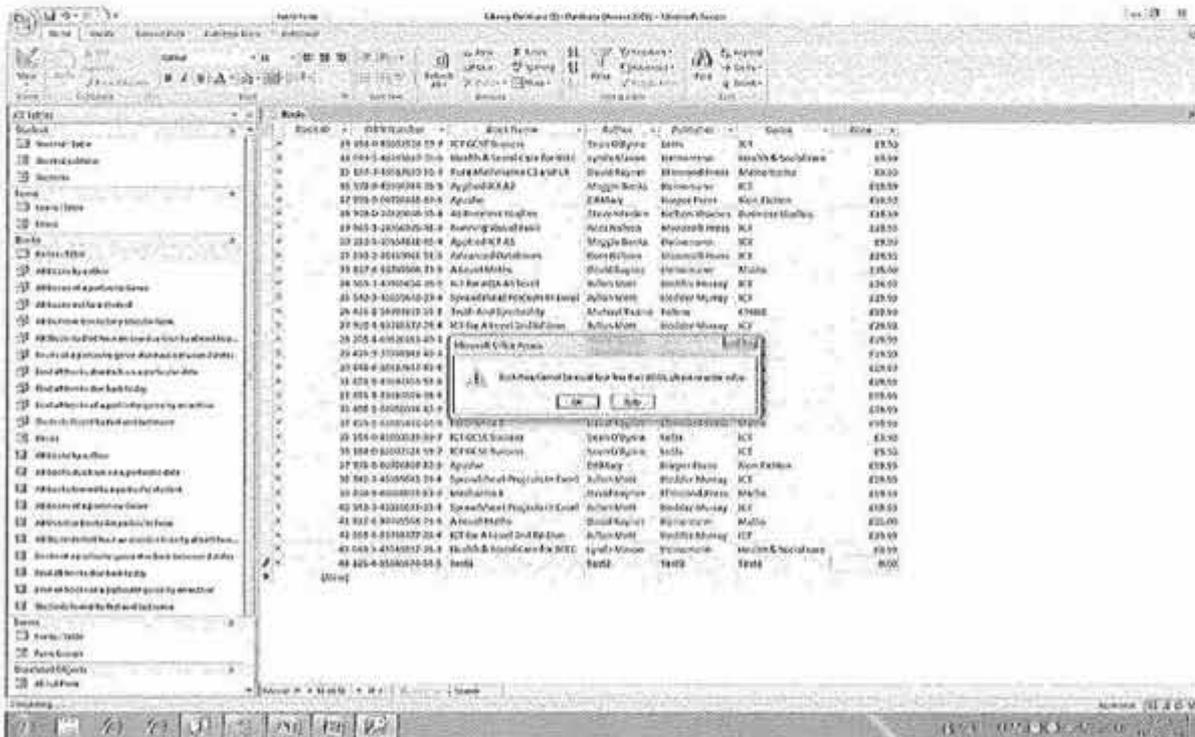
the table being visible and only numbers can be entered into these fields. The book name, Author, Publisher, Genre and Price fields can all also be seen to be working with the data I have input being clearly visible.

### Test 25)

(+)	39	488-9-84058034-83-7	Mechanics 1
(+)	40	542-3-45355643-23-4	Spreadsheet Projects in Excel
(+)	41	837-4-92749568-73-9	A Level Maths
(+)	42	968-4-93748377-28-4	ICT for A Level 2nd Edition
(+)	43	043-5-46245837-36-5	Health & Social Care for BTEC
(+)	44	125-4-54	

I have performed this test in order to ensure that the Input mask for the ISBN number is working correctly. After testing this input, the mask is clearly working correctly as numbers have to be the correct length and are automatically hyphenated to the correct format. This input mask is therefore working correctly as the only error that can be caused in this field is user error where the end user inputs the incorrect number.

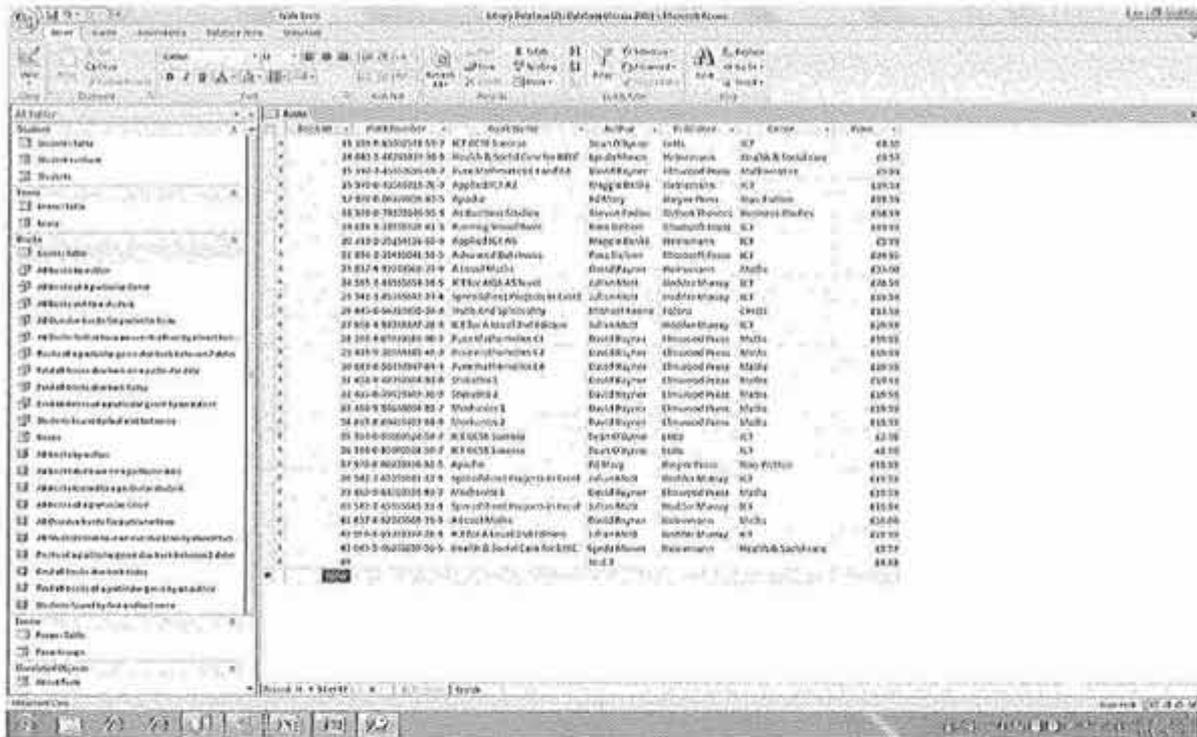
### Tests 26 and 27)



This screenshot shows tests 26 and 27 working. After entering the value 0.00 the database automatically changes it to £0.00 and therefore the currency format on the table is clearly working correctly. Once I have tried to enter this selection I have also had an error message

produced saying that the field value cannot equal £0.00 and that I must re-enter the value, meaning this field cannot be left blank by the user of the database.

**Test 28)**

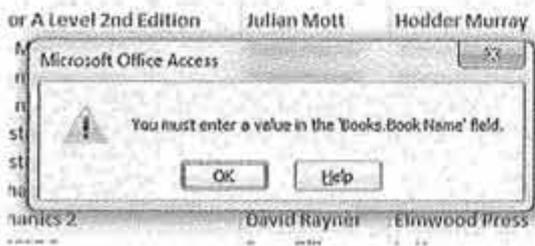


This screenshot shows that it was possible for me to enter a record with only the "price" and "Author" fields entered, with all other fields left blank. This table should require all fields to have data input before a record can be entered; therefore I have now looked at how to fix this error.



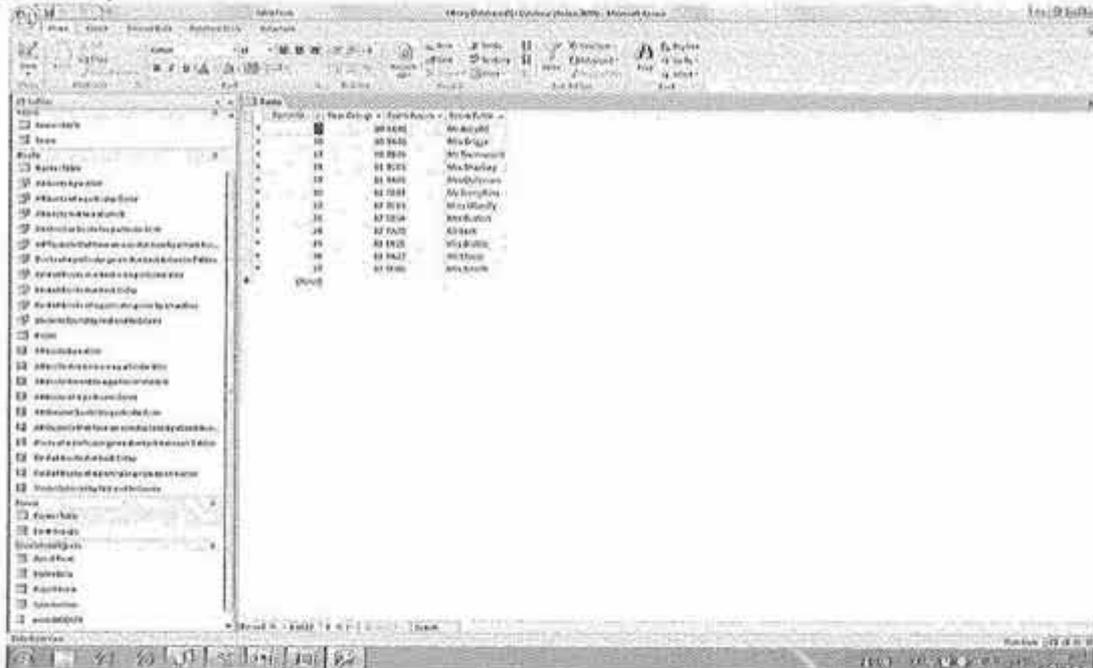
After investigating this within the design view for the table, I noticed that although the "required" option was set to "yes", the table

would allow the field to be of zero length, meaning that it could be left blank. I therefore set this option to "no" so that the user was forced to enter data into each one of these fields. After doing this for multiple fields, I then re-tested this to ensure that all fields now needed to have data input into them.



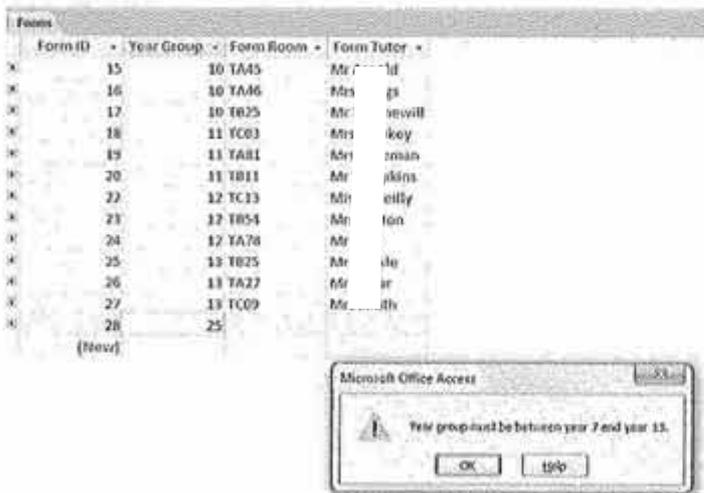
Upon changing this value in the table design view, I was now produced this error message when the record was entered without a "Book name" value being input. This was tested for all fields within the database which all now worked correctly meaning data had to be input into them all.

## Tests 29-33)



This screenshot shows the tests being performed to ensure that data can be input into the table which can be easily seen once entered. This test has shown me that data can be input and therefore the Table is working correctly and how it was designed to operate. I am now going to continue looking more in-depth at this table in order to ensure that the validation rules that I have set up are operating properly now I am sure that the table I have set up is working correctly.

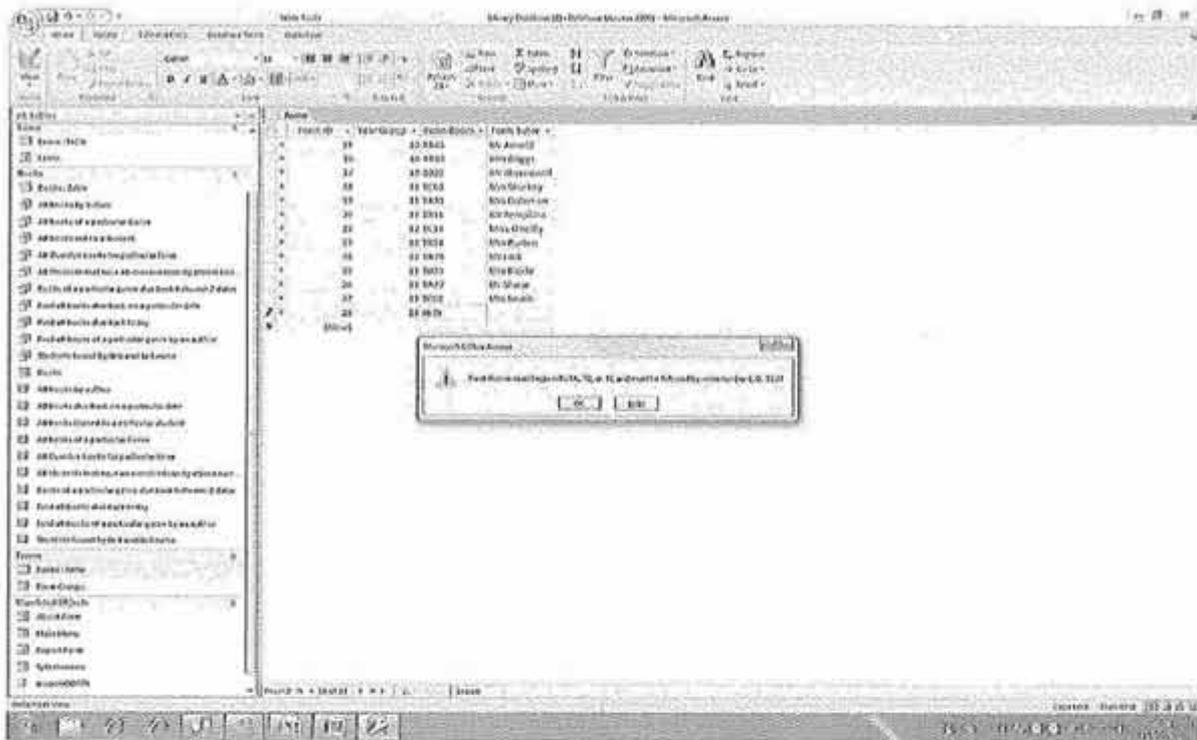
## Test 34)



This test was done to ensure that the Year group that was entered could only be between years 7 and 13 as these are the year groups available within the school. After entering any number such as "25", I was produced the customer error message saying "Year group must be between 7 and 13". This shows that this validation rule is therefore working correctly, preventing inaccurate information from being entered into the database. After entering year groups

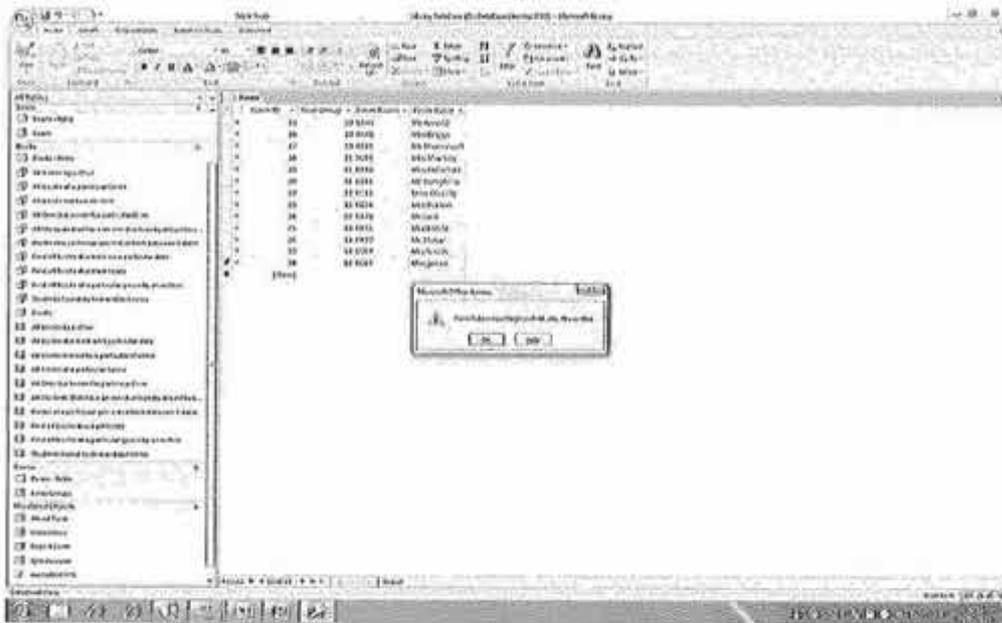
between 10 and 13 I can also see that no error message is produced meaning the validation rule is also allowing correct data to pass.

## Tests 35 and 36)



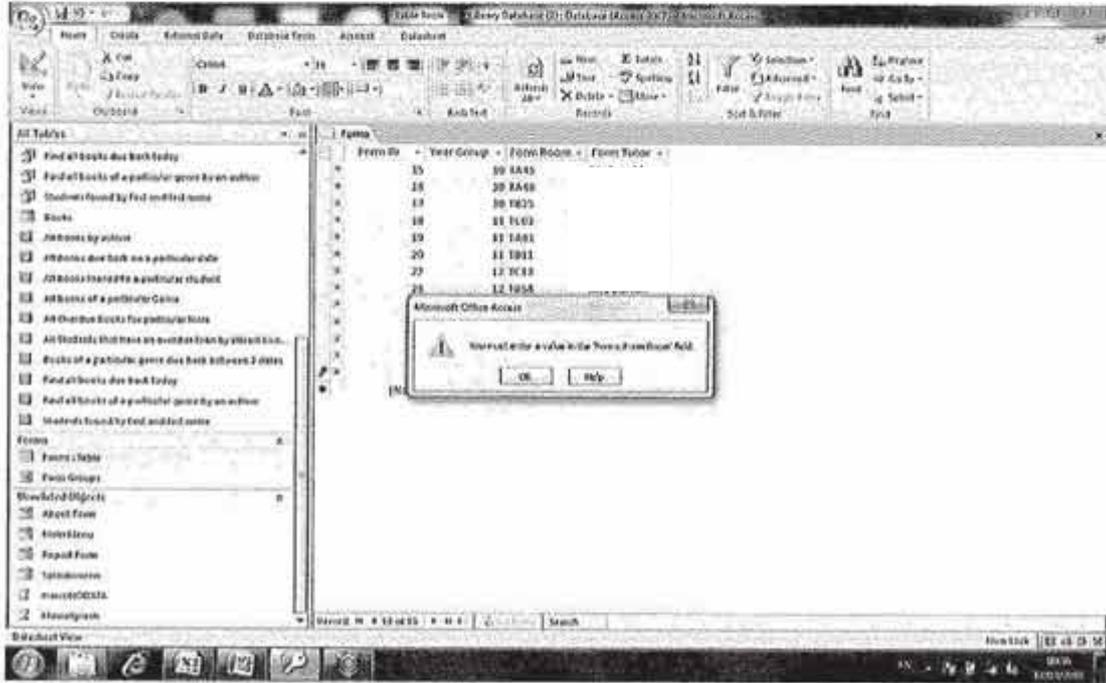
These tests have been performed to ensure that the validation rules and input masks for the "Form Room" are working correctly, again to prevent incorrect data being entered. The input mask is working correctly by only allowing me to enter things which are formatted Letter, Letter, Number, Number. After also testing the validation rules that I have used by entering "JB79" an error message was also produced because it did not start with TA, TB or TC. This therefore shows that the validation rule that I wish to use is working correctly. It can clearly be seen from the above screenshot after entering data in the correct format E.G. TA45, the database allows the data to be input with no problems.

### Test 37)



I then moved on to test the validation rule to ensure that the start of a Form Tutors name was entered correctly, again helping to prevent errors being made in the database. Unfortunately a Tutors name could be anything therefore the only validation I can make that the name must start with Mr, Mrs, Miss or Ms. After entering some text such as "Mss" which could be done as a typing mistake, the validation rule worked by not allowing this input showing the error message "Form tutors name must begin with Mr, Mrs, Miss or Ms", therefore showing the validation rule is working correctly. It can also be seen from this screenshot when entering correct data the validation rule is allowing the data to be input and therefore is working the way it should, as it is not stopping correct data from passing.

### Test 38)



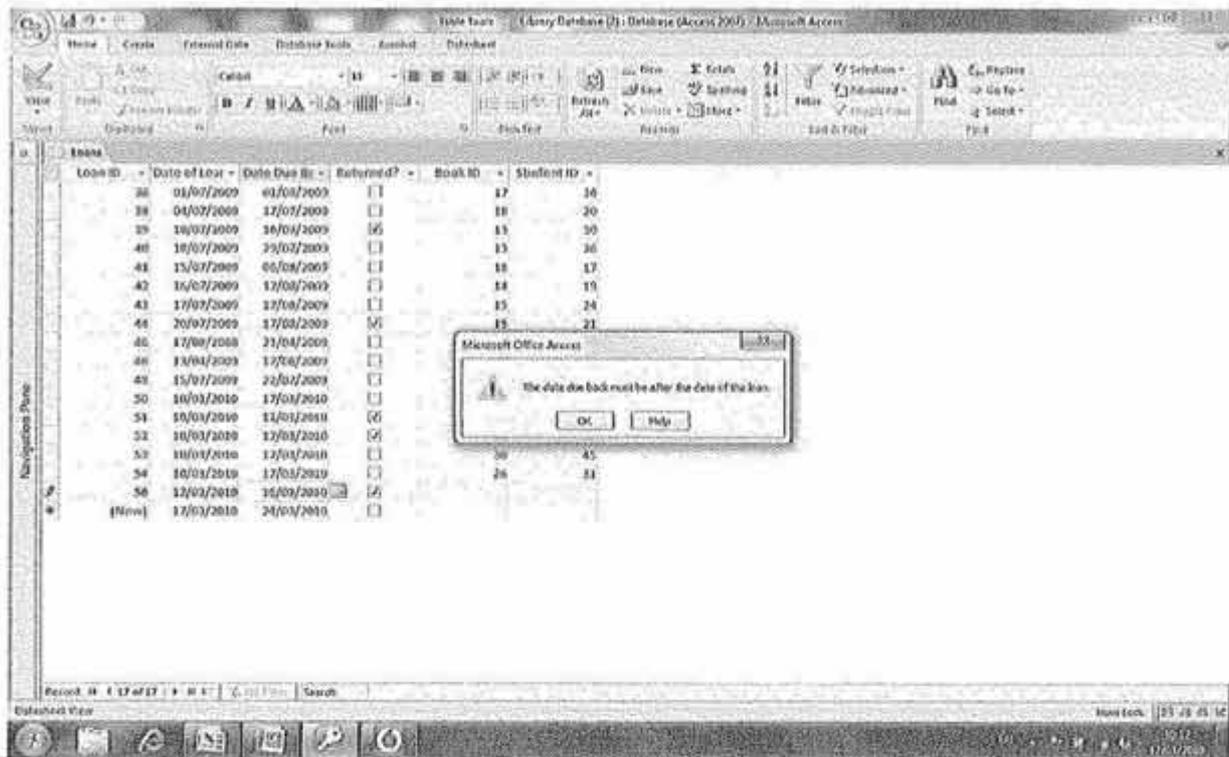
This above test has been done in order to ensure that the user has to fill in all fields on a record before it can be added to the database in order to prevent incorrect or incomplete information being given. After testing each of the fields within this table I had to ensure that each field would complete before I was able to proceed, otherwise an error message such as the one shown above was produced. This therefore shows that this is working correctly and that the end user cannot add a record which is incomplete.

### Tests 39-44)

Loan ID	Date of Loan	Date this is due	Returned?	Book ID	Student ID
37	01/07/2009	01/08/2009	<input type="checkbox"/>	13	16
38	04/07/2009	17/07/2009	<input type="checkbox"/>	18	20
39	10/07/2009	16/07/2009	<input checked="" type="checkbox"/>	13	30
40	16/07/2009	23/07/2009	<input type="checkbox"/>	33	28
41	15/07/2009	06/08/2009	<input type="checkbox"/>	16	17
42	16/07/2009	17/07/2009	<input type="checkbox"/>	24	19
43	17/07/2009	17/08/2009	<input type="checkbox"/>	15	24
44	20/07/2009	17/08/2009	<input checked="" type="checkbox"/>	19	21
46	17/08/2009	23/04/2009	<input type="checkbox"/>	16	18
48	17/04/2009	17/04/2009	<input type="checkbox"/>	31	17
49	15/07/2009	23/07/2009	<input type="checkbox"/>	16	19
50	10/01/2010	17/01/2010	<input type="checkbox"/>	25	40
51	10/01/2010	17/01/2010	<input checked="" type="checkbox"/>	42	46
52	10/01/2010	17/01/2010	<input checked="" type="checkbox"/>	15	42
53	10/01/2010	17/01/2010	<input type="checkbox"/>	30	45
54	10/01/2010	17/01/2010	<input type="checkbox"/>	26	31
(New)	17/01/2010	24/01/2010	<input type="checkbox"/>		

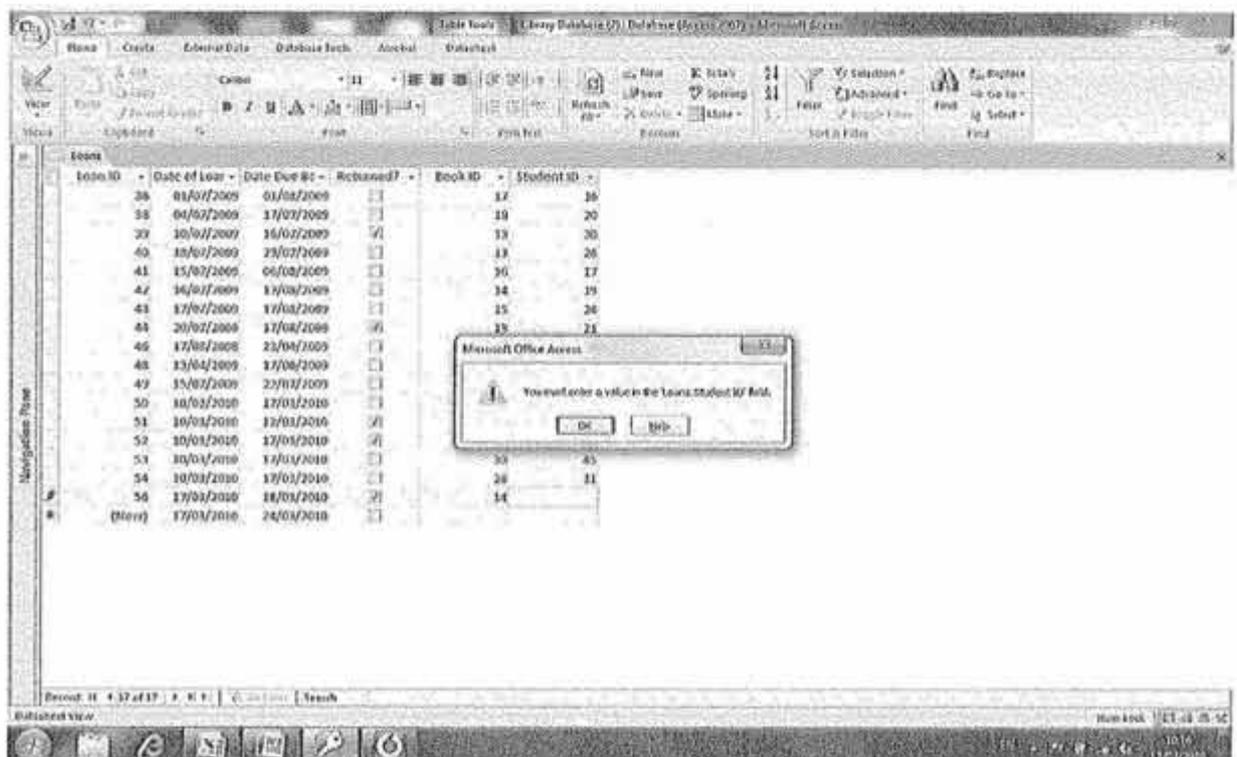
I have now entered data into all fields of the database in order to ensure that the data can be entered and that it can be clearly seen. After performing these tests I can see that my Loan ID is working correctly and that this is automatically numbered when a new record is added. Once again this is unique and therefore will only appear once within the database. I can also clearly see the date of the loan and date that the loan is due back is also formatted correctly, with the data being clearly displayed in a format dd/mm/yyyy. I can also see that the 'returned' check box is also working along with the Book ID and Student ID. As a book will have been loaned out more than once and students will have loaned more than one book I have also ensured that these entries can be entered multiple times, therefore I have input Book ID "13" twice and Student ID "17" which the database has clearly allowed.

### Test 45)



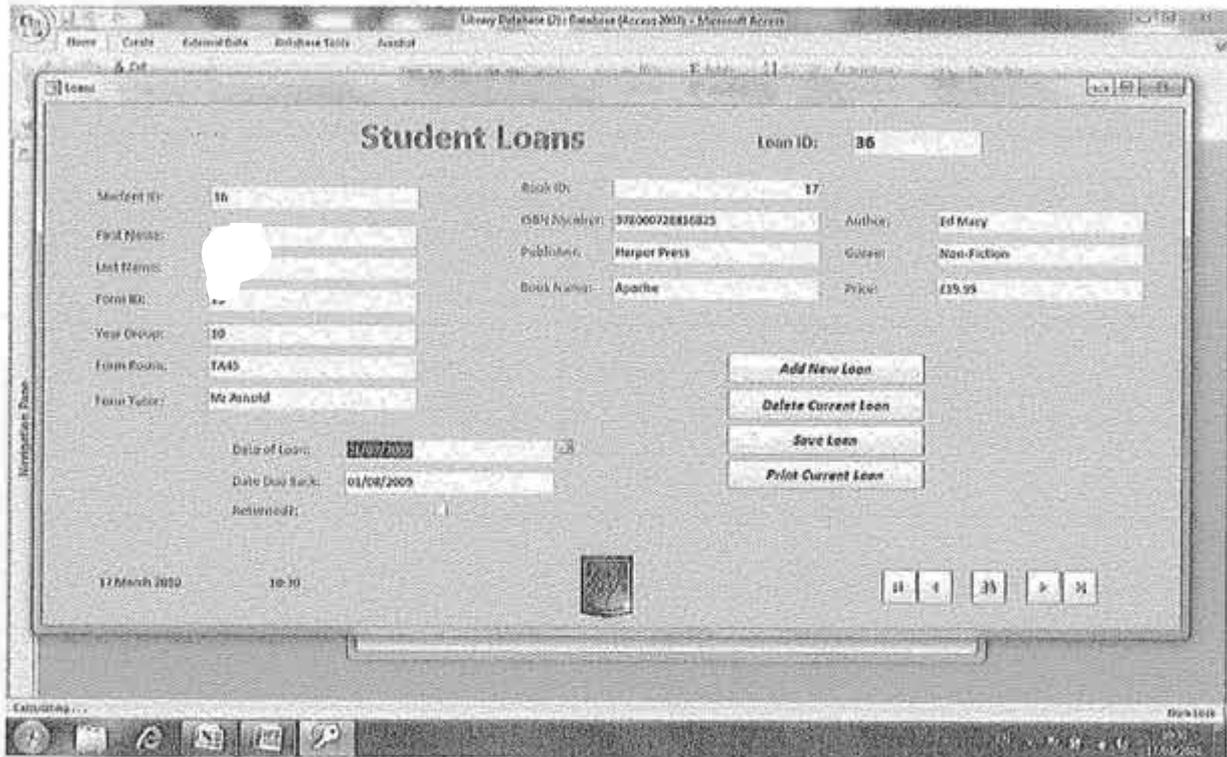
This above screenshot shows where I have now entered a date due back which is 1 day before the date of loan being made. When this is entered, the above error message displays "the date due back must be after the date of the loan". This again prevents incorrect data from being entered into the table as this may cause errors at a later date should the date due back be set before the date the loan was made as this is clearly not possible.

### Test 46)



After working through this table I can see that all fields must be completed before a new record can be added to the table. This once again prevents incomplete records being accidentally added to the database as this may cause problems when returning the book. If a field is not completed, an error message such as the one shown in the above screenshot is again displayed to the user alerting them that they need to enter a value in that particular field.

### Tests 47-49)



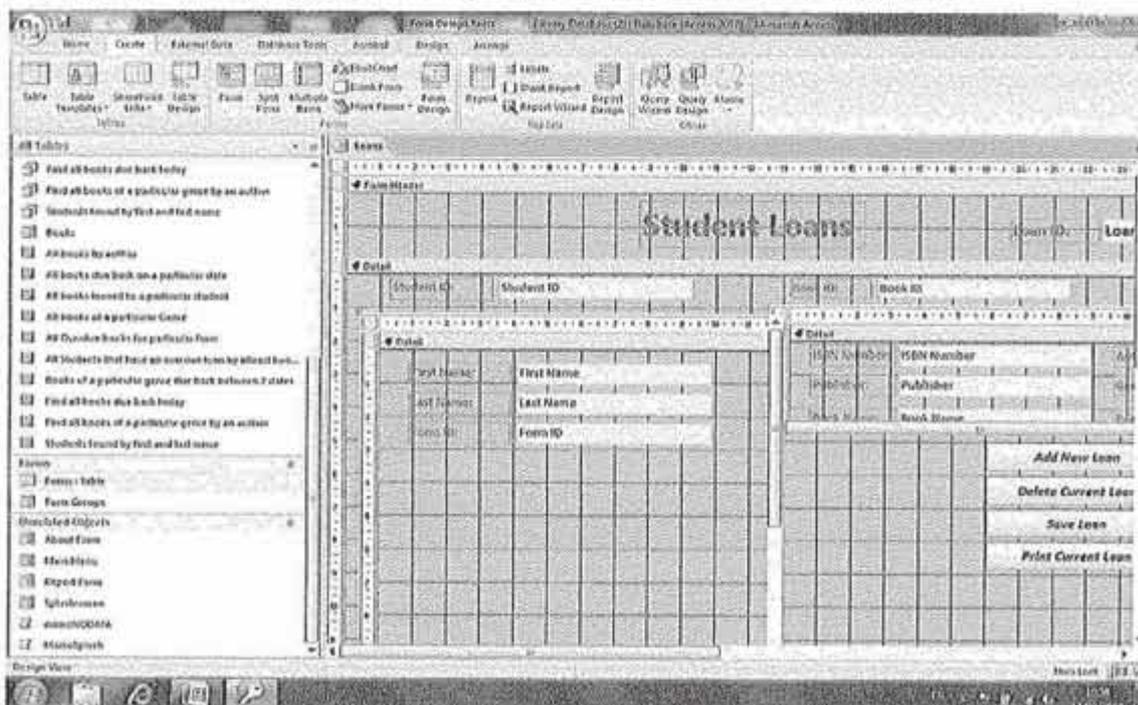
This screenshot is simply to show that the Student Loans form opens correctly once the button within the main menu is selected. It is also to ensure that the data is displaying correctly and that the time and date are working and updating. This clearly shows that the form is opening correctly and that the data can be clearly seen. I can also see that the time and date are displaying correctly however do not appear to be updating from when the form is launched.

After further investigation into the problem with the clock, however the page is showing "calculating" every second. This means that the time is updating but does not display unless the page is refreshed, such as navigated to a different record. The only way to overcome this would be to use another macro which would force the page to refresh every minute, however this would mean should this happen whilst the user is working on the database, anything not saved each minute may be lost. I have therefore decided to leave the time and date as they are as this may otherwise cause further problems.

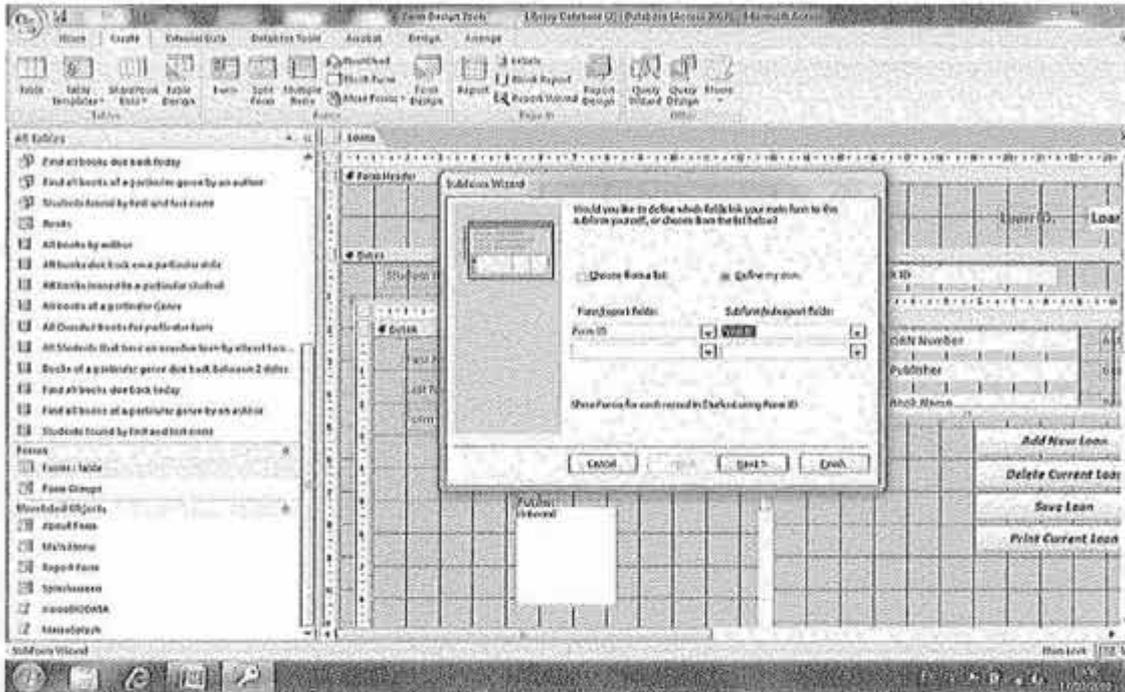
I have however noticed another two faults with the form. Firstly when different records are selected, the "year group", "form room" and "form tutor" are not updating to the newly selected record meaning incorrect information is shown. As shown in the below screenshot, this student now has form ID 23 however information is still being displayed for form ID 15. Finally the data shown for "Book ID" is out-of-line which also looked unprofessional; therefore I am going to change the alignment on this.



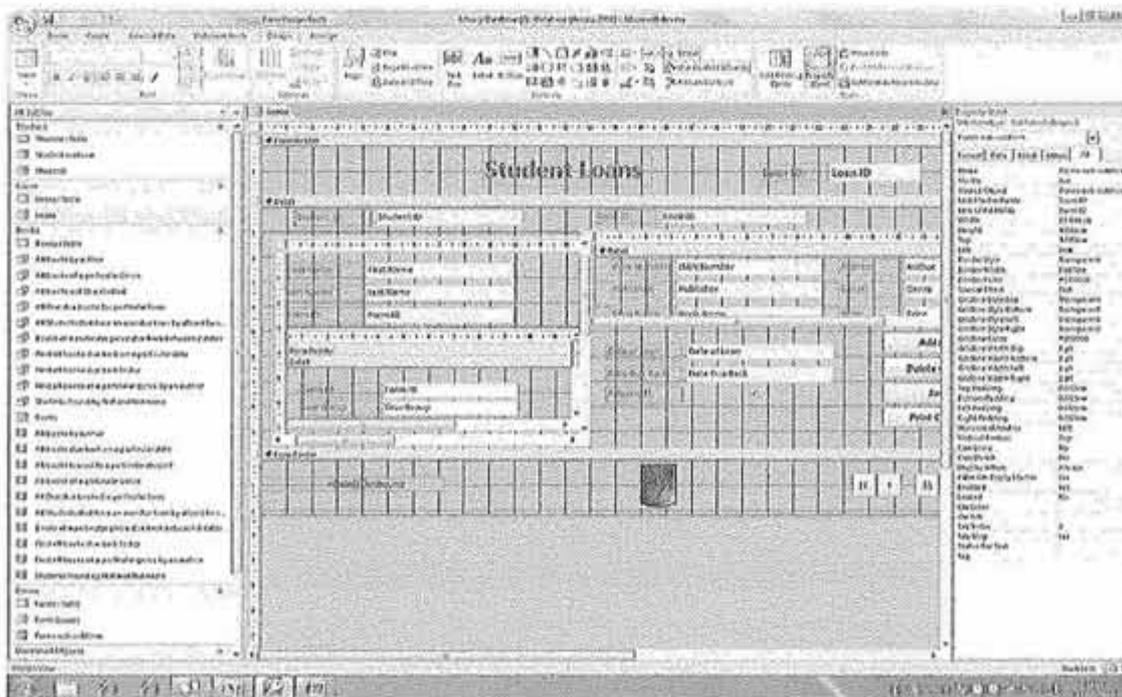
I am now going to go back into design view for my database and re-create the "Form groups" sub form to add into this form once again so that the data should update when different records are selected. I will then hide duplicated fields such as "form ID" as this will be generated from both the "Forms" table and from the "Students" table.



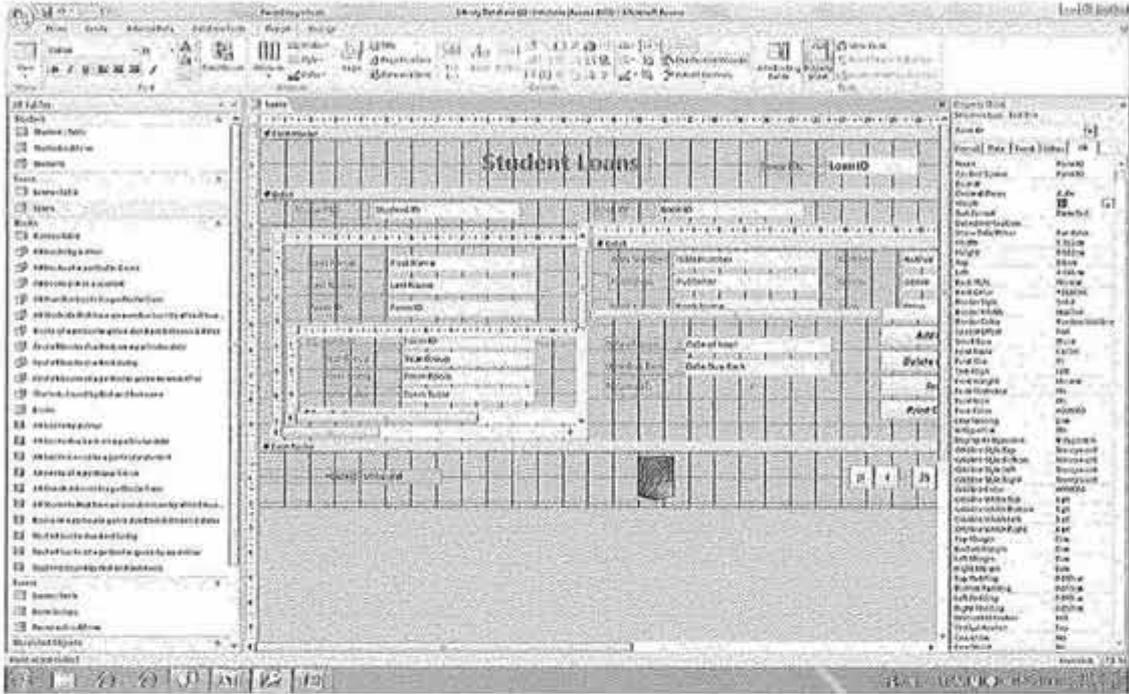
This screenshot now shows the design view for the Form that I have created. In order to link in the "Form groups" table correctly, I am going to have to create a sub-form within the "student" sub-form. I have therefore begun by deleting the current entries within the "Student loans table" and extending the size of the "students" sub-form. I then selected the "Form Groups" Table and dragged this into the "Students" sub-form.



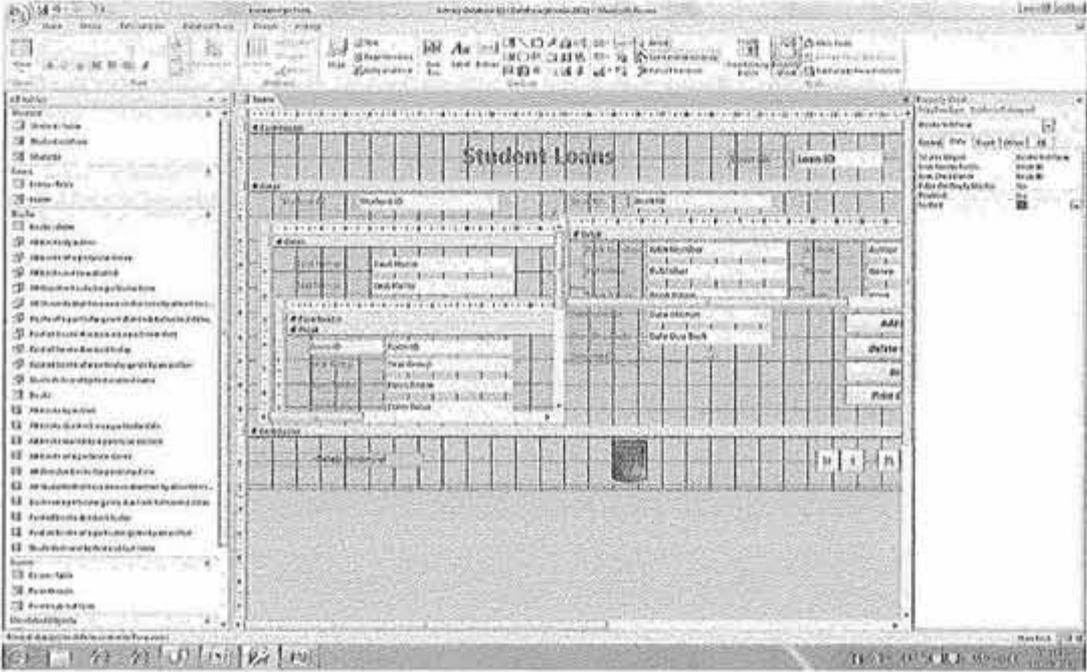
After entering this table I was given the option to define a link from the sub-form to the current form that I had made. I therefore selected the two "Form ID" entities within both the "Students" form and the "Form groups" form in order to link the two together.



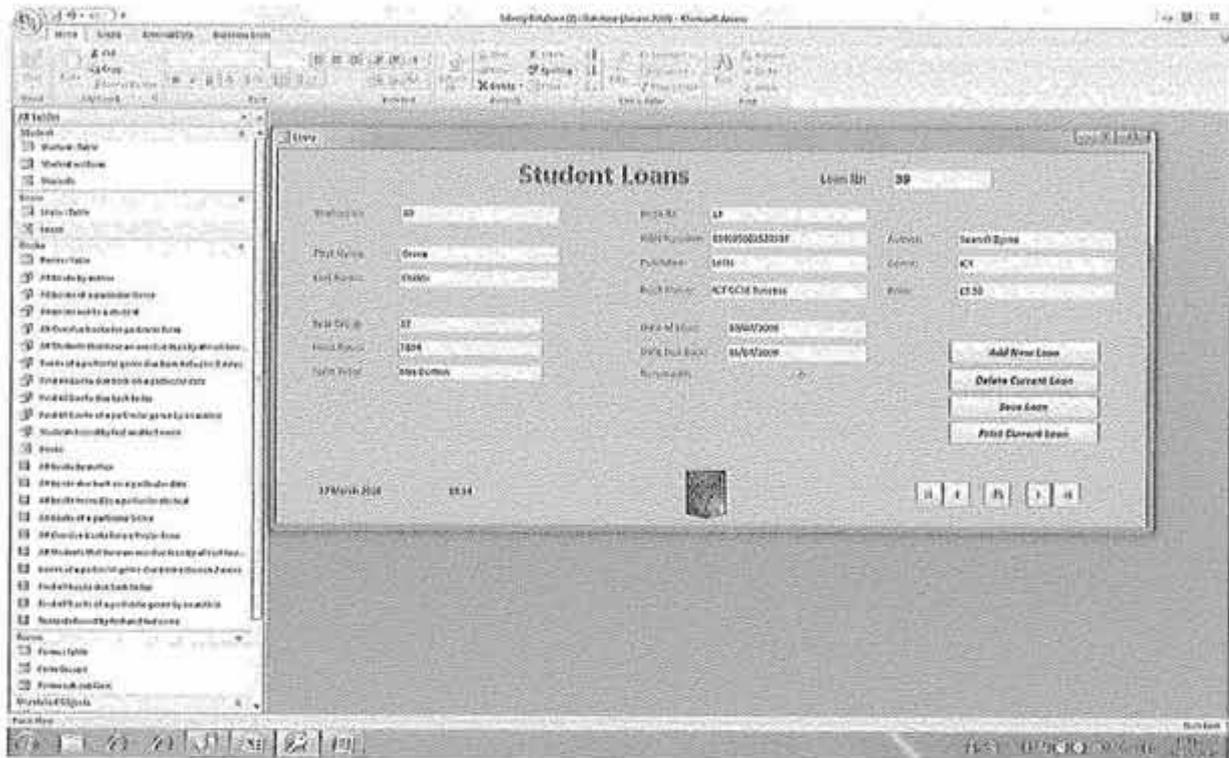
This above screenshot now shows the new sub-form that I have created with the relevant fields inside. I have also used the properties sheet for this sub-form in order to ensure that the sub-form is not visible and only its contents can be seen. I have therefore disabled all views other than "Single Form" view and removed the border and pre-set navigation options.



This above screenshot simply shows me no setting the second "Form ID" field so that its property "visible" is set to "no" meaning it can be seen within design view and therefore the link exists yet when shown on the form, this field will not be displayed. I have done this as it is already going to be displayed once within the "students" sub-form.

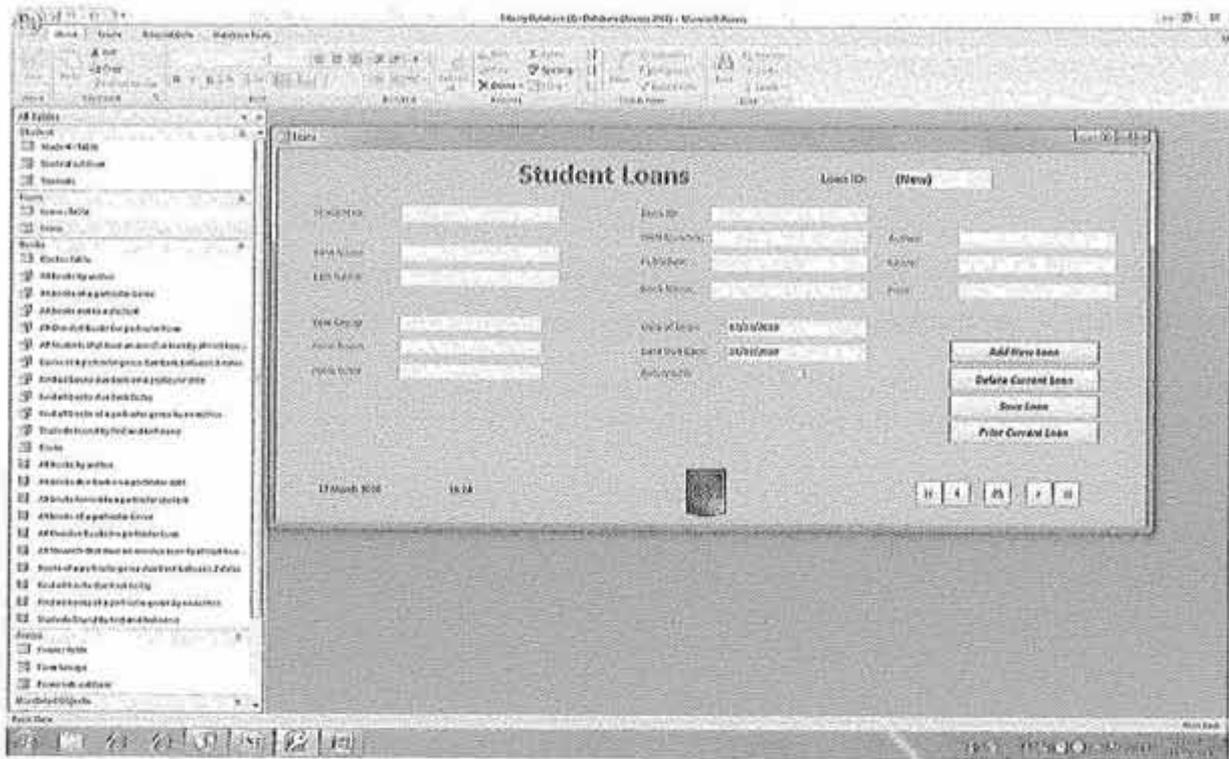


The next screenshot now shows the fix for another error that I found whilst fixing the previous problem with the database. This is that when the “Student Loans” form was open, changes could be made to other tables such as information about “books” and “students”. This should not have been possible as I have provided other forms within the database for this and by allowing these changes to be made, accidental changes are more likely. I have therefore selected the option “locked” for both sub-forms meaning data within these cannot be changed. Finally I have also selected the “Book ID” field and selected the left align option in order to align text the same within this field as all other fields.

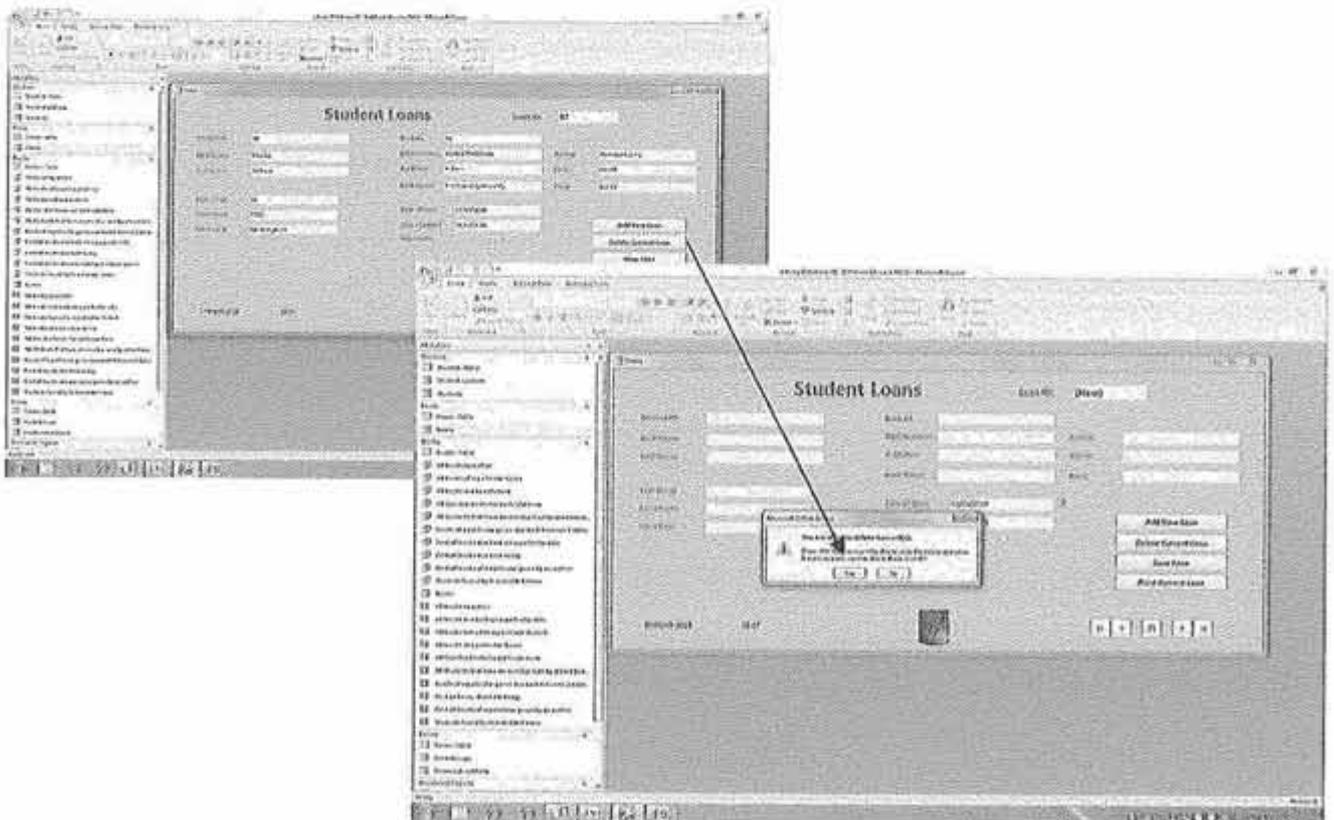


This above screenshot now shows the “student loans” form after I have made the numerous above changes. All text is now left aligned and clearly visible, with the “year group”, “Form Room” and “For tutor” attributes now updating when a different record is selected. I have also tried to select data within the sub-forms and I am now not able to edit this.

Tests 50-53)



Upon the "Add new loan" button being selected a blank form is produced where I can input a Book ID and Student ID to recall details along with the date of the loan and the date that it is due back.



After entering a new loan I then also used the "Save button" in order to ensure the loan was saved to the database. This was working correctly as the record was saved once the database was closed and re-opened. I then accessed this loan (ID 57) and selected the delete button. Upon selecting this, a warning message was produced confirming that a record was about to be deleted. I firstly selected "No" and the record was kept. I then repeated this process selecting "Yes" where the record was then deleted successfully.

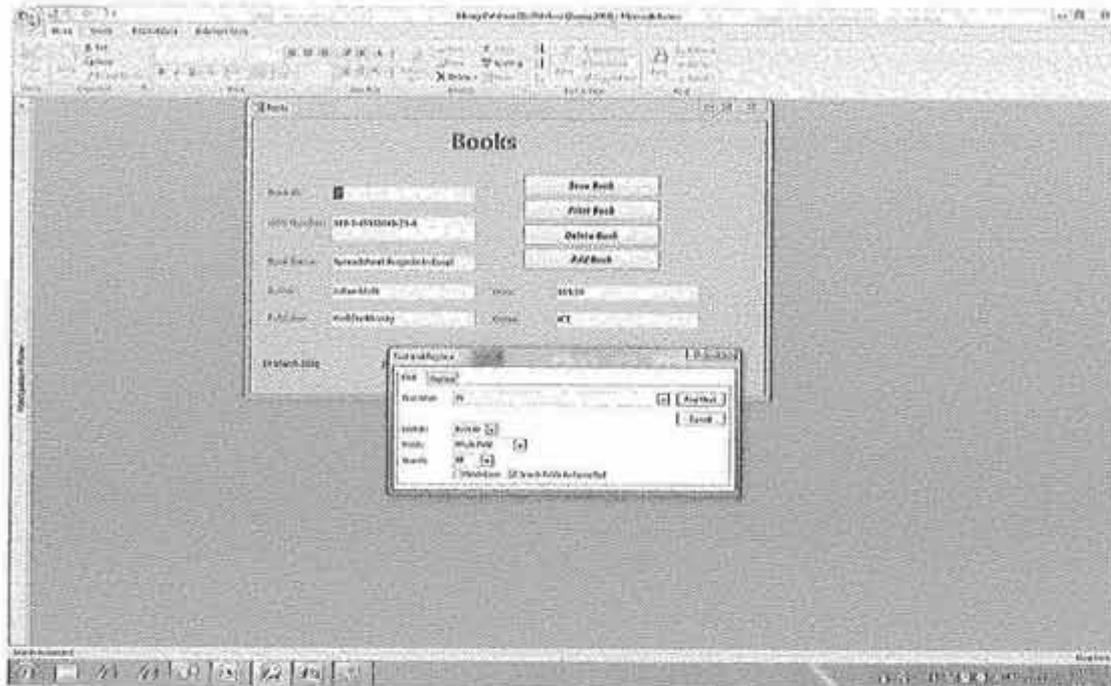


I then also tested the "Print current loan" button in order to ensure that this button was working as intended. After this was selected I was then given the print dialogue where I had the option to print the record, therefore this button is also working correctly.

Finally I then also tested the record navigation buttons that I had added to the forms. These buttons all performed correctly directing the user to the previous, next, first and last record. Upon the search button being selected I was produced the following search option:

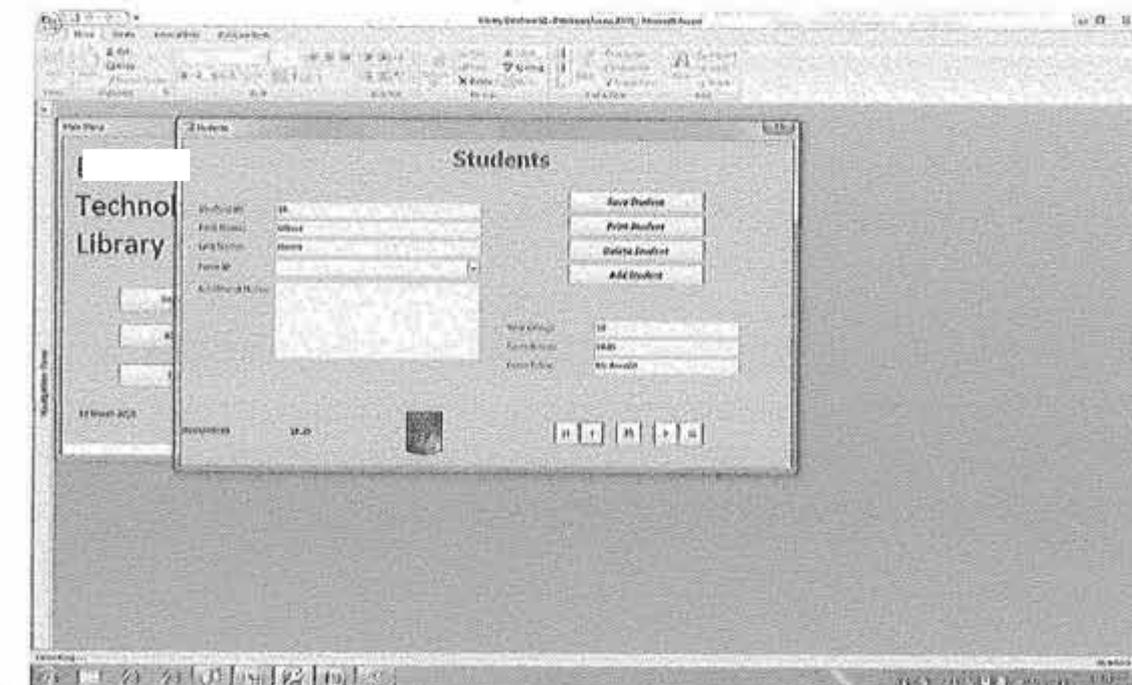


Upon selecting the Print button I was therefore presented with this dialogue box again where I had the option to change print settings before sending that record to the printer. After selecting the Delete button I was also produced an error message warning the user that the delete operation cannot be undone once the record has been removed. These buttons provided for use on the form are therefore working correctly.

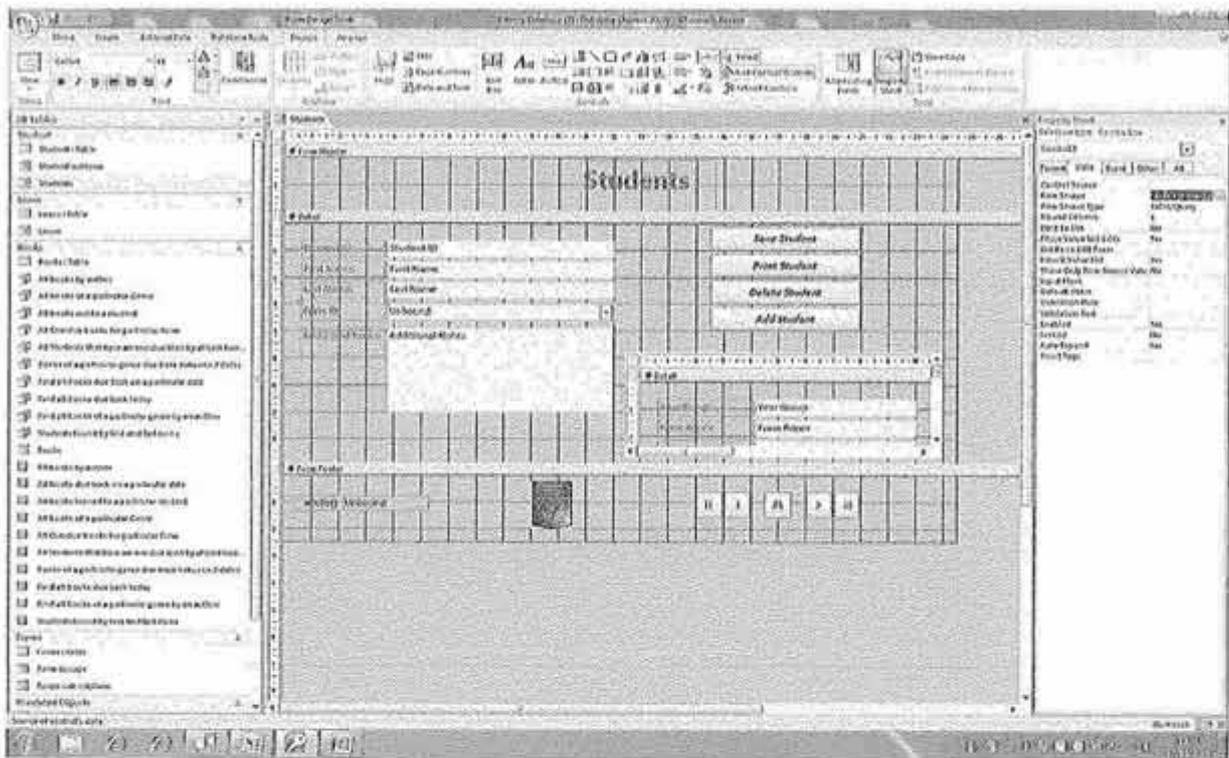


The final test for the "Books" data input form was to again test the navigation options between records. Once again these navigation options allowed me to select the first, last, previous and next record as well as search for a particular record. After selecting the search button and inputting find ID "25", the record for Book ID 25 was successfully displayed.

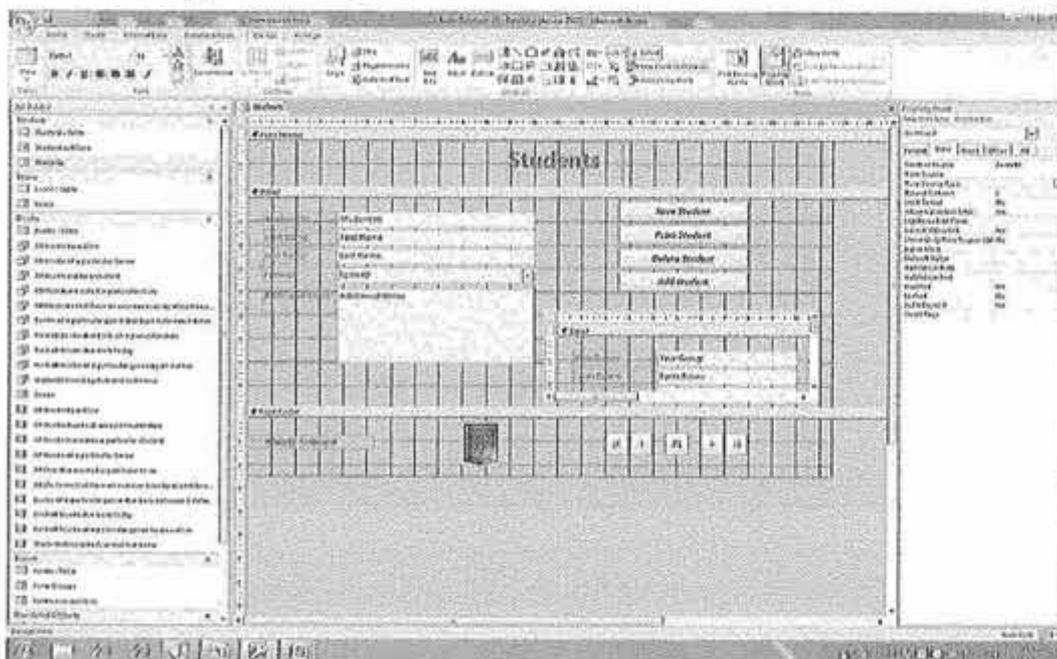
### Tests 63 and 64)



After selecting the button from the main menu Test 63 was successful as the Students Form opened correctly from this button. Upon opening it, I did however find that the drop-down box to select "Students" from was not operating correctly as it was not displaying any values for any records selected.



The first thing that I tried was to edit the 'Row Source' as this had a value which looked at another table in order to produce the list. After removing this, the drop-down box also stopped working and I therefore had a blank field within the form which was shown as



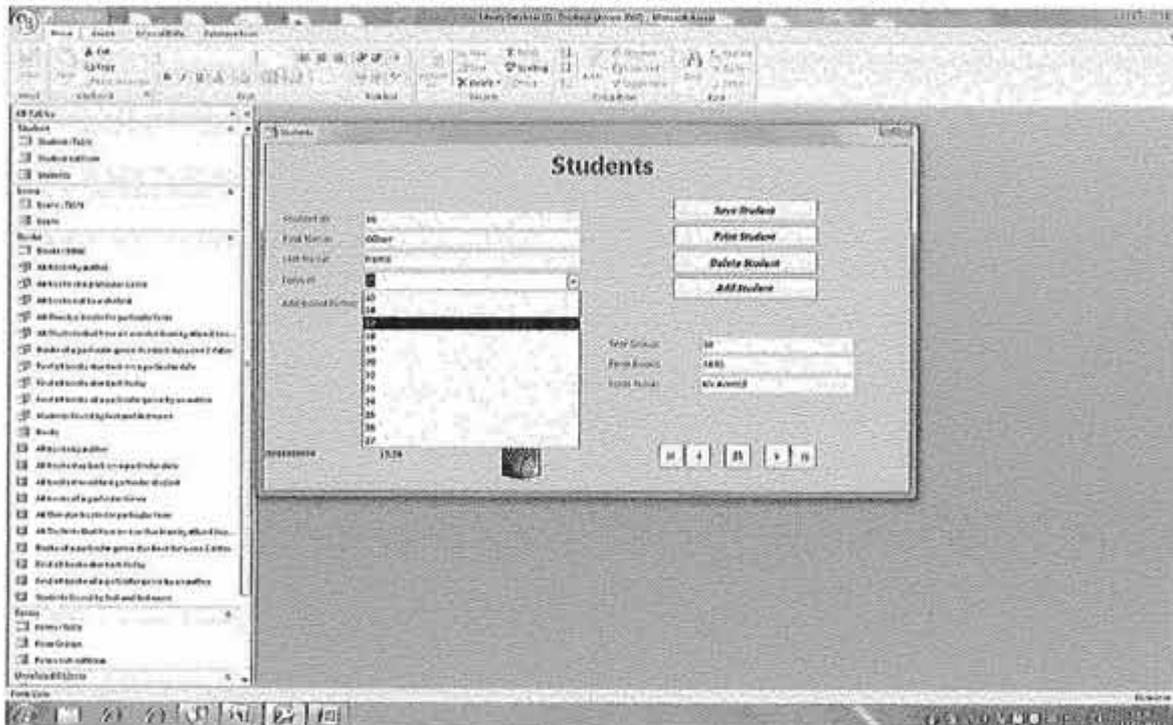
"unbound", meaning it was not linked to any part of the table. To add it in order to show the correct data I then went into the "control source" which shows where the source data is located within the table. I therefore then changed the control source value to be set to "Form

ID" which then linked this field to the "form ID" part of the table. The sub-form showing

“Form group” information was working throughout this and therefore did not require editing. After reviewing the changes I had made, the form ID was now shown within the data input form, however when adding a new record my drop-down box was still not functional. I therefore added the following code back into the “Row Source” to define which values could be entered into this table. This code used the “form groups” table in order to display the ID numbers for each form group available, meaning if a Form Group is added or deleted, it would be added or removed automatically from this list:

**SELECT [Forms].[Form ID] FROM Forms ORDER BY [Form ID];**

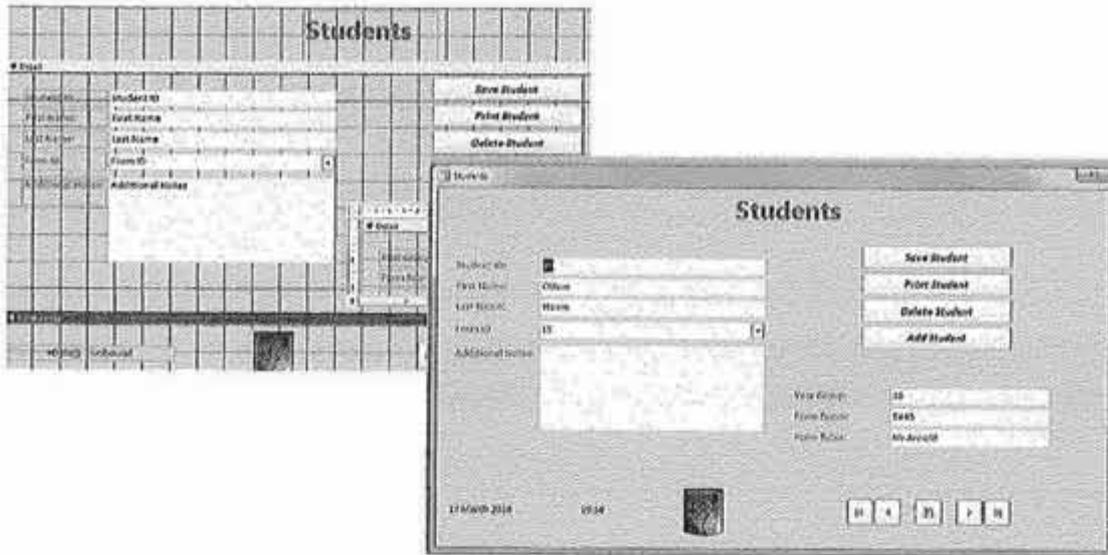
This code within the Row source was telling the database to look for the “Form ID” values that were available from the “Forms” table. It also specifies that these results should be ordered by Form ID meaning they will be displayed in numerical order.



This screenshot now shows the drop-down box working with the values displayed correctly within the form now that I have fixed the Form ID field.

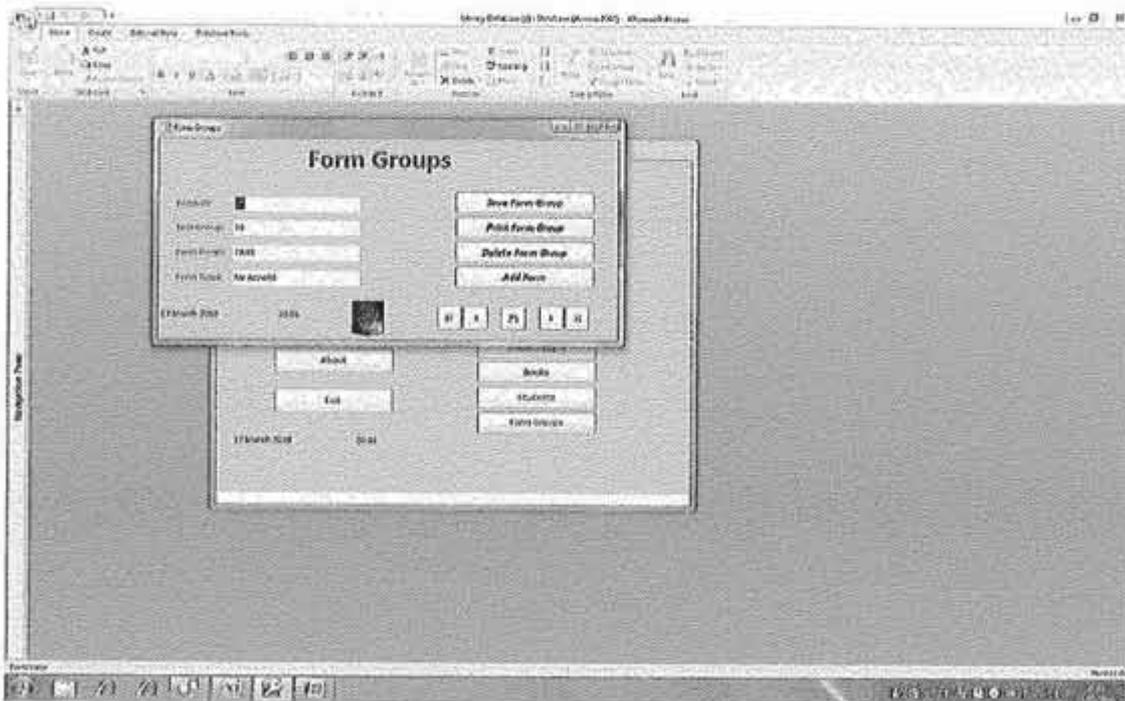
### Test 69)

As can be seen from previous screenshots, the time was displaying correctly however the date is shown as “#####”, clearly showing that there is a display error on the form. After entering design view the fault was found to be that the field was simply not large enough to fit the date within the space and therefore causing this error to be made.



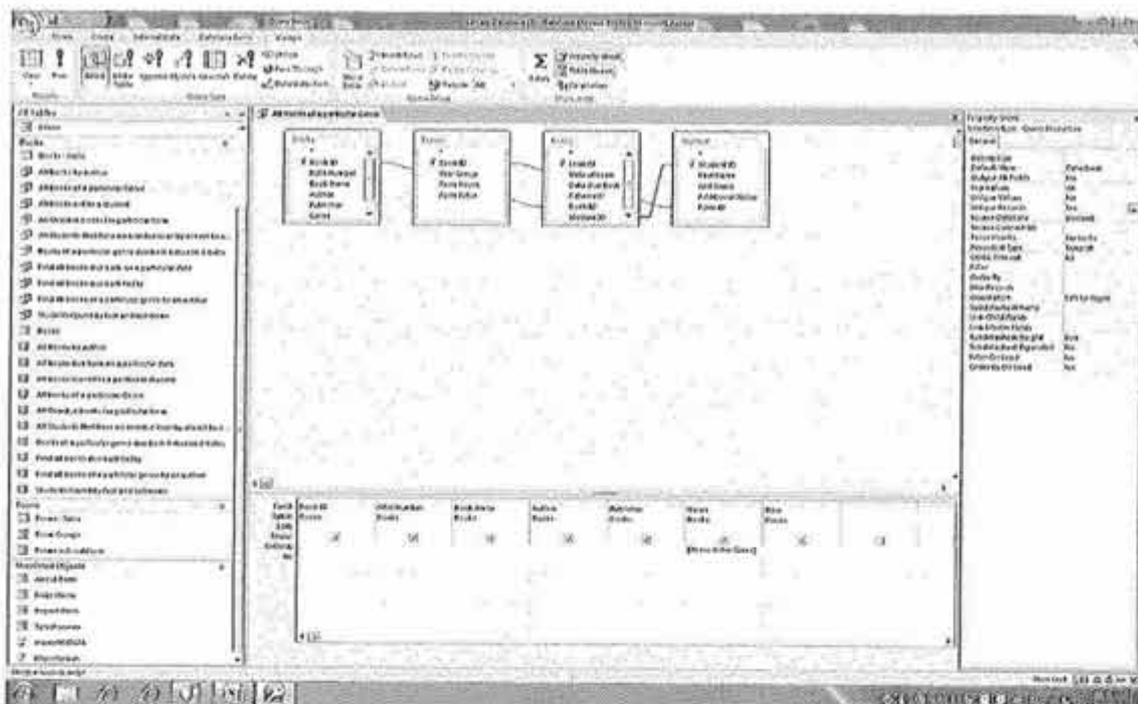
It can now be clearly seen after increasing the size of this field, there is now enough room to display the date and it is now showing correctly when run rather than the error of ##### being shown.

### Test 71)



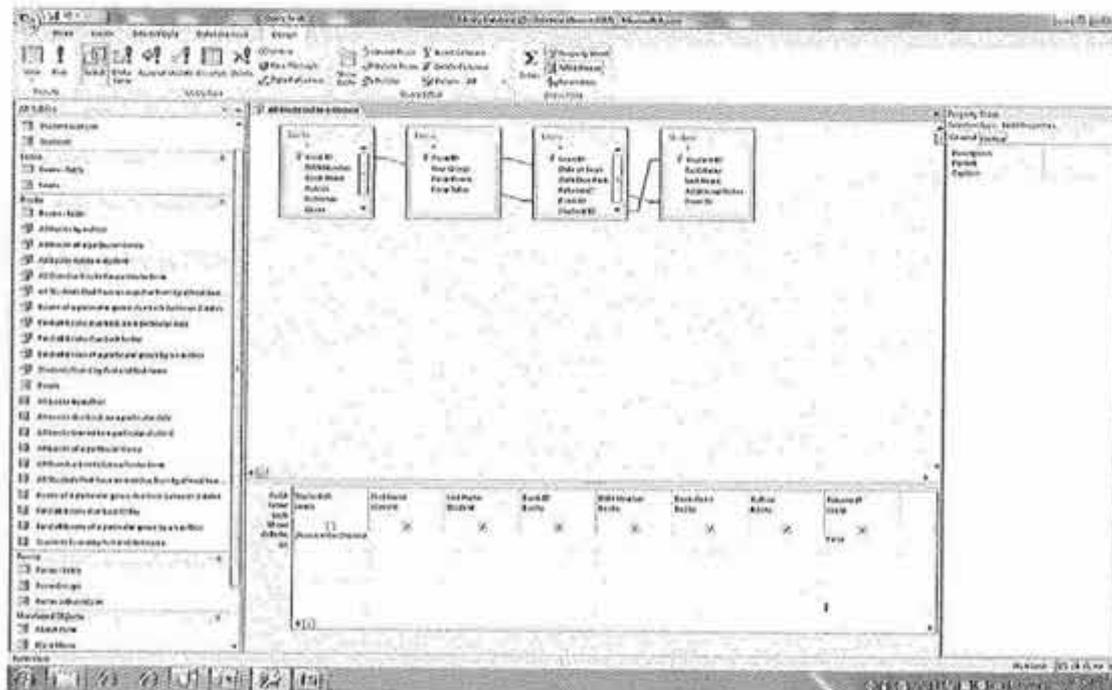
This test was to ensure that the button from the main menu opened the Form correctly and that all data was displaying clearly. As can be seen from this screenshot, data can be clearly seen and also updates when a different record is selected.

### Test 79)



The query that I have run "Display all books of a particular genre" is returning multiple results for the same book, possibly because they have been loaned more than once. I therefore went into design view for the query and into the properties sheet. I have then selected the "Unique records" option to be "Yes" meaning they will only appear once, no multiples of the same record will be displayed. After running the query again, I can now see that this is running correctly and the required results are being displayed.

### Test 80)



Correct results were displayed within the query however an "expr1000" field was displayed and the "student ID" field was displayed. Neither of these are required. Upon entering design view the "Expr1000" field was generated because two parts of the query were "Student ID" fields and therefore could both not have the same name. I therefore removed the one that was not required. I also deselected the "show" option on the other "student ID" field as the user will know this as it is the input value, meaning only relevant data is now shown.

**Test 83)**



This query is running correctly however when the dates are entered into the parameter they have to be manually types into the correct format. This is fairly hard to remember and means that errors are likely to be made or the End user may not be able to use the system efficiently. Besides making a separate form to input two dates which can be linked to the parameter query there is no way to specify an input mask or format on the pop-up box from a parameter query, therefore I have now simply changed the text in order to specify the format it must be entered in.



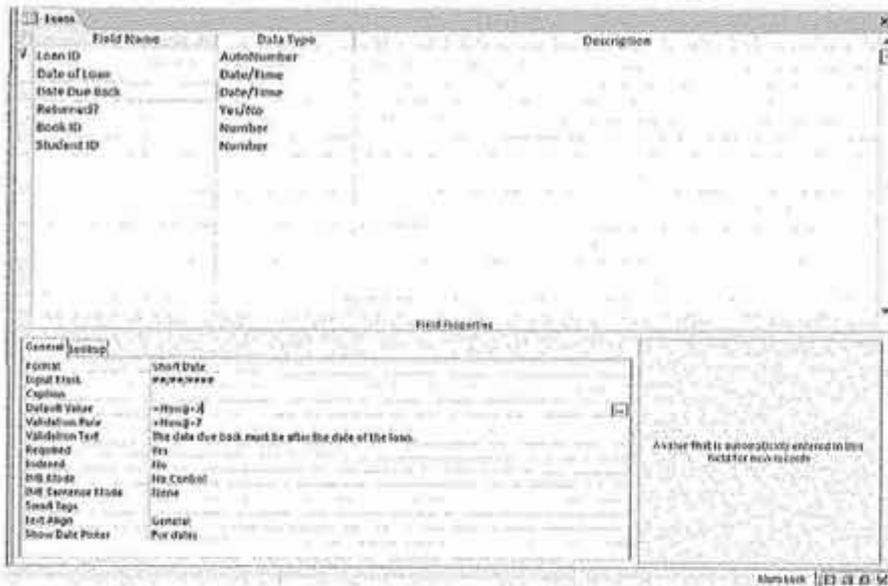
**Test 84)**

Test 84 is in order to ensure that the query "All books due back on a particular date" is working correctly. Upon running this query there was a fault found and results were not being displayed correctly (A blank results page was being returned). After examining the query design there was not a fault with this, therefore I then looked at the values that had been entered within the "loans" table for a possible reason why the database could not find the dates entered. Upon looking at this I found that despite setting the "format" of the cells to be "short date" therefore in the format DD/MM/YYYY, if the date picker provided was used, the time stamp was also added after this date within the cell, meaning the query

59	18/03/2010	25/03/2010	<input type="checkbox"/>	23	29	could not find the date as it also wanted
61	18/03/2010	25/03/2010	<input checked="" type="checkbox"/>	17	23	
*(New)	18/03/2010	25/03/2010 13:59:55	<input type="checkbox"/>			

the exact correct time to be entered.

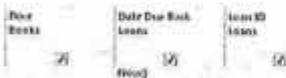
In order to correct this, I have decided to use an input mask as well as the current validation rules so that only the date can be displayed and not things such as the time afterwards. This means that data weather typed or input by the date picker will be entered in the correct format and therefore should allow the query to work. I then also updated the displayed parameter query message so that end users know how data needs to be input.



This screenshot shows the changes that I have made in order to ensure that data within the table is now working correctly and can therefore be recognised by the database. After now re-running the query I am now displayed the correct result and therefore the

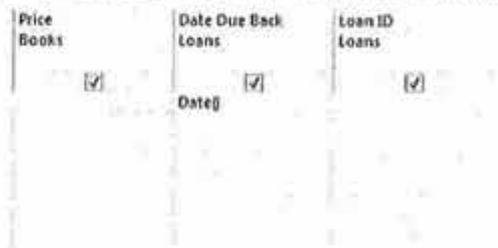
problems with the dates have now been fixed.

### Test 85)

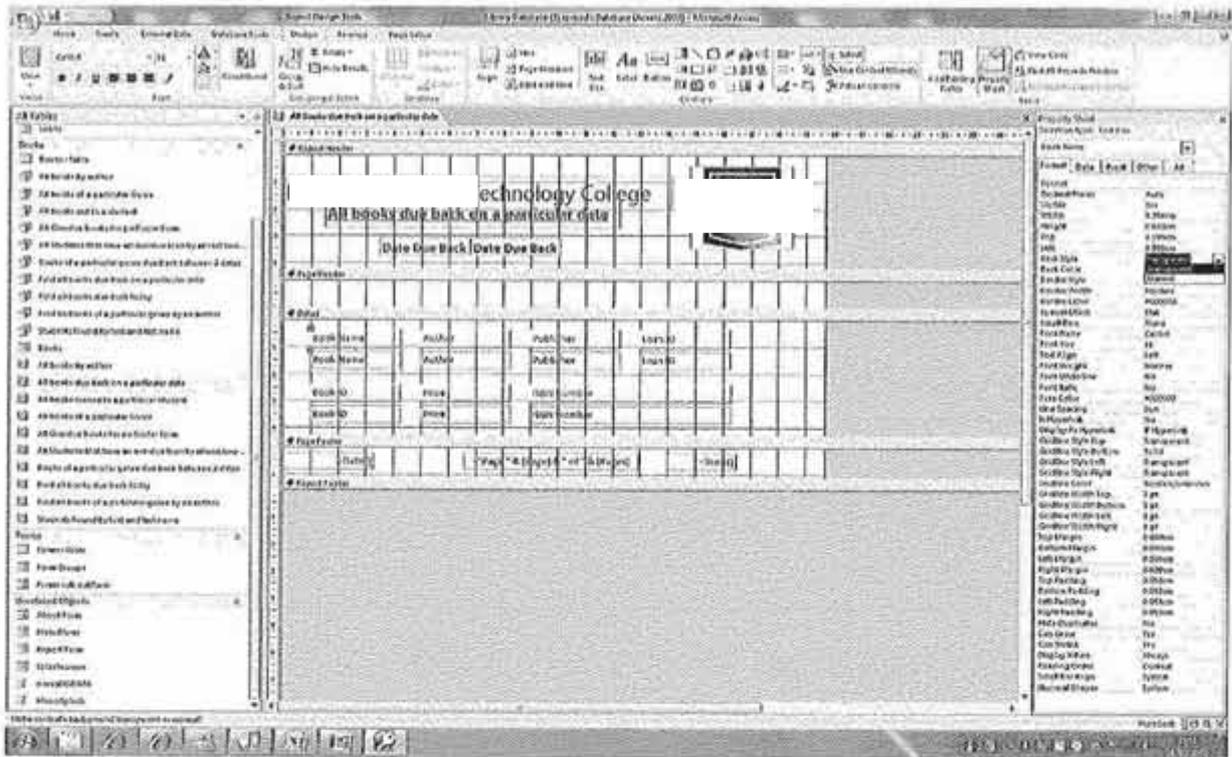


This test is in order to ensure that the query "All books due back today" is working correctly. This may previously have not worked however I have now edited the input mask for the dates in the above test so should now also work correctly. After testing the

query I was again provided a blank list and therefore this query was also not working the way it should have been. After looking at the query design, I again thought that this could be similar to the issue above. After looking into this I found that the "criteria" that I had used was "=Now()" which was calling the system Date and, once again, Time which was causing the problem. I therefore changed the criteria within the report to =Date() which simply uses the system date and not the time. After changing this, the report was now working correctly.

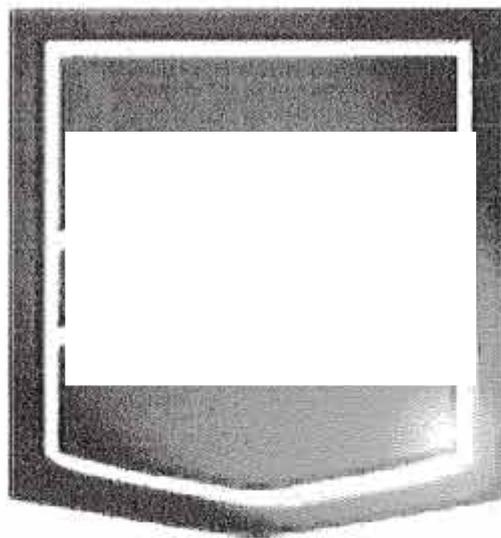


Test 94)



This test showed when printing the reports that white text boxes were shown across the background colour which looked very unprofessional. I have therefore gone back into the design view for the report and changed the "Back style" for each text box to "transparent" so that the box cannot be seen.

# Library Database System



## User Guide

Last Updated 03/03/2010 for version 1.01

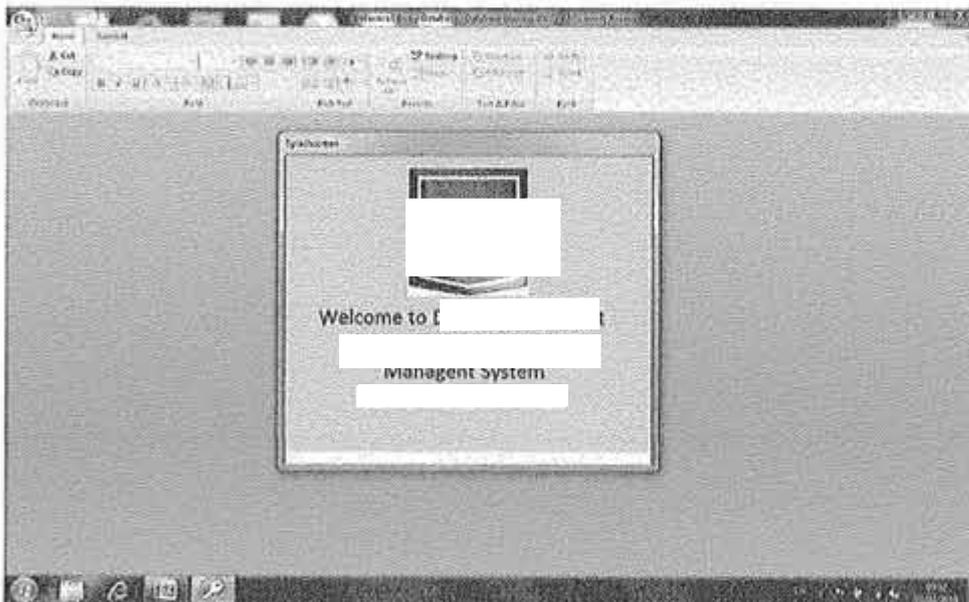
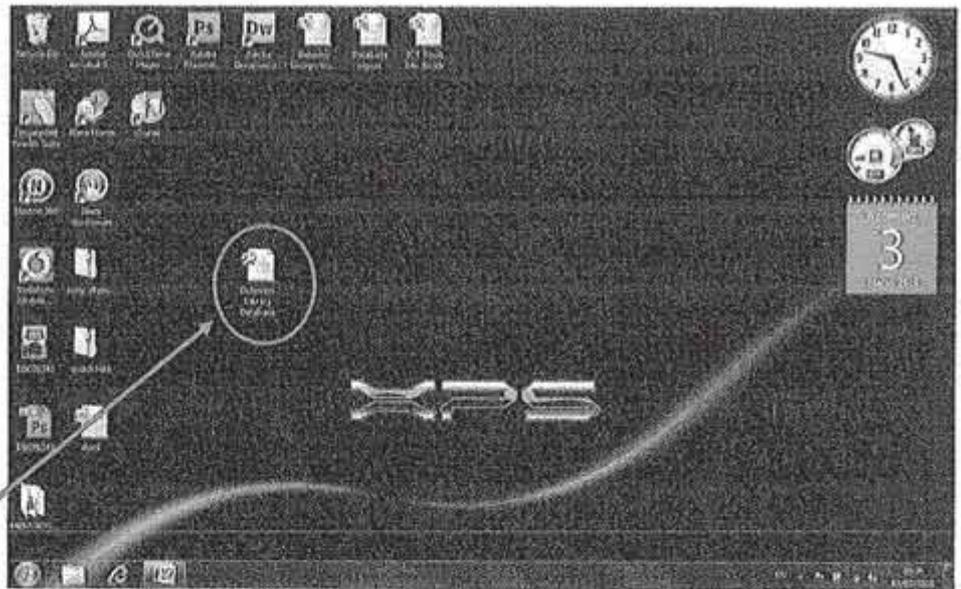
## Contents Page

<b>Starting the Database</b>	<b>3</b>
<b>The main menu</b>	<b>4</b>
<b>Form Navigation Options</b>	<b>6</b>
<b>Student loans</b>	<b>7</b>
<b>Books Menu</b>	<b>10</b>
<b>Student Menu</b>	<b>12</b>
<b>Form Groups Menu</b>	<b>14</b>
<b>Reports Menu</b>	<b>16</b>
<b>Printing Reports</b>	<b>18</b>
<b>Backing up the database</b>	<b>19</b>
<b>Acceptable data and Validation rules used</b>	<b>21</b>
<b>FAQ (Frequently Asked Questions)</b>	<b>23</b>

## Starting the database

You will find after booting up your PC and logging on to windows that there is an icon named "Library Database" on your desktop. This is the file that must be run in order to open the database. To open the database you can therefore double click on this file and the database will open.

This is the file you must open.



Upon opening the database you will see a welcome screen open to show you that you are opening the database. Please be patient, you do not need to click anything and after a few seconds this screen will disappear. This database has been locked so that you can only edit content using the forms provided,

therefore you should not worry about trying to enter or access data using other methods that shown in this user guide, as all other features have been disabled to increase the databases security.

## The Main Menu

Below is a copy of the main menu that will be showed shortly after the database is loaded. This menu is in effect the "control panel" for the entire database where all tasks can be accessed from. For your ease, all record options are on the right side of this menu under the title "edit details". You must select these options to Add, Edit, Print or delete any record stored in the database.

To Add, Edit, Delete or Print a loan that is stored in the database, select the option "Student loans"

To Add, Edit, Delete or Print a record for a Book that is stored in the database, select the option "Books"

To Add, Edit, Delete or Print a student that is stored in the database, select the option "Students"

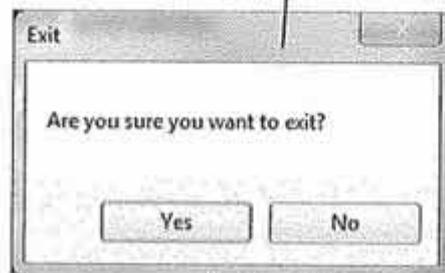
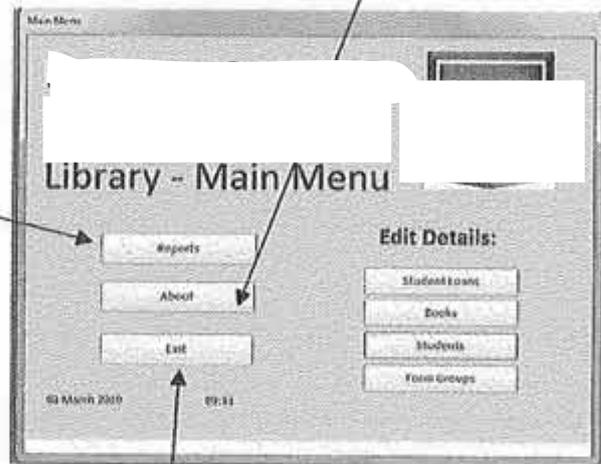
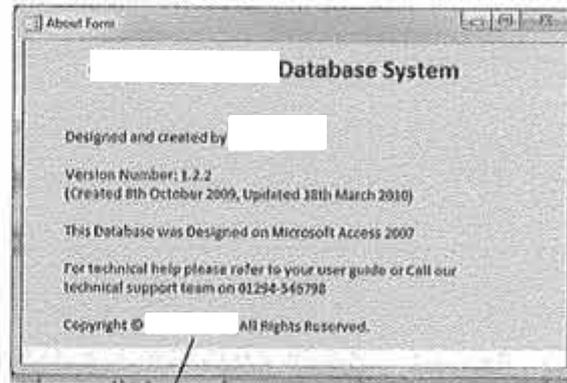
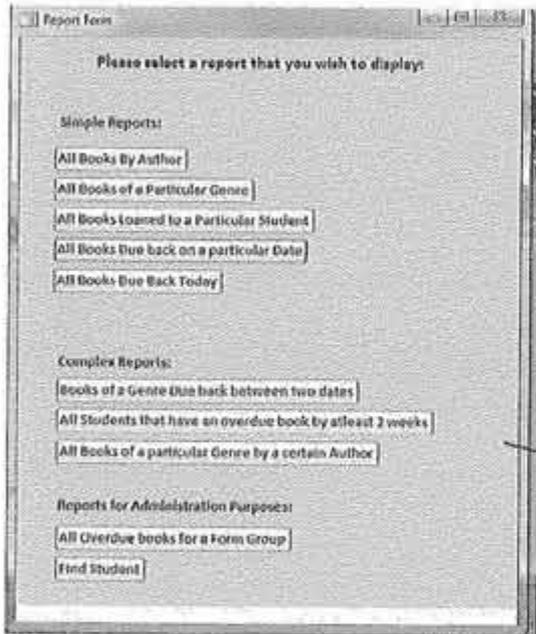
To Add, Edit, Delete or Print a record for a Form group that is stored in the database, select the option "Form Groups". This option is only likely to be used each year if a form group changes for example has a different room or teacher.

You should NOT attempt to change details other than the ones for which the button and menu is designed. For example, within the "students" option, the linked "form group" is displayed. You cannot change details about that "form group" within the "students" menu

The other options that are shown on the main menu are the options for "reports", "About" and "exit". These have a number of different uses and are displayed on the left side of the main menu.

The "About" button can be used to display information about the database. This contains information such as who it was created by, what version the database is running and when this was last updated.

The "reports" menu option allows you to access a list of reports that can be produced by the database, such as a list of all students with an overdue book. A more detailed explanation of this menu is detailed later in the guide.



The exit button must be used to close the database. It is important that you must use this to close the database correctly before shutting down the database, otherwise you risk corrupting the database meaning all data could be lost. It is therefore recommended that you make regular backups of the database on an external device should this happen. The details for doing this are shown later in the guide.

For the input forms that are detailed on the right side of the main menu, there are a number of navigation features that are used on all four of these forms and perform the same option throughout the database.

This next button is used to navigate to the previous record stored within the database, for example if you were on record 25, this would navigate you back to record number 24.

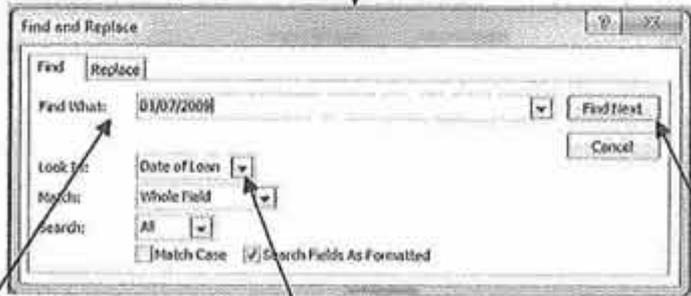
This option is used to navigate to the next record within the database. For example if you were again on record 25, this would navigate you to record number 26.



This option is used in order to navigate to the first record that is used within the table. It will therefore navigate to the first available loan, student, book or form group that was input into the database.

The next option is a search facility if you wish to search for a particular ID number. By selecting this button a message box will pop-up asking you what you wish to search for.

This final button is used to navigate you to the last record within the database. This means if you close the database and wish to access the last record you input, you can select this option to do so.



Enter the value or data that you wish to search for here

Enter which part of the form you wish to look in

Select the "Find next" button to begin the search.

## Student Loans Menu

The next part of the database is the “student loans” data input form that has been created for you to input loans into the database. As has already been detailed in this guide, this form can be accessed by selecting “Student Loans” from the main menu.

After typing in a “Student ID” number, all other student data will be automatically displayed.

After typing a “Book ID” number, data for that book will be shown.

A new loan ID will be generated each time a new loan is added to the database.

The screenshot shows a Microsoft Access application window titled "University Library Database - Database (Access 2007) - Microsoft Access". The main form is titled "Student Loans" and contains the following fields and controls:

- Student ID:** 10
- First Name:** [Redacted]
- Last Name:** [Redacted]
- Fees ID:** 15
- Year Group:** 10
- Form Book:** TAAS
- Form Tutor:** Mr Arnold
- Book ID:** 17
- ISBN Number:** 978000728816825
- Publisher:** Harper Press
- Book Name:** Apache
- Author:** Ed Mary
- Genre:** Non-Fiction
- Price:** £19.99
- Loan ID:** 36
- Date of Loan:** 11/03/2009
- Date Due Book:** 02/04/2009
- Returned?** [Checkbox]

Buttons on the right side of the form include:

- Add New Loan
- Delete Current Loan
- Save Loan
- Print Current Loan

At the bottom left, the date "03 March 2010" and time "10:51" are displayed. At the bottom right, there are navigation buttons (Home, Back, Forward, End) and a status bar showing "Page 1 of 1" and "03/03/2010".

Enter the date that the loan is being made and the date that the book is due back here.

Select this box once the book has been returned to the library.

Other options – see pages 8/9

Navigational options (Discussed on page 6)

This input form ties all of the data that is available from the database together. Once the loan input is completed, it will automatically be provided a "Loan id" so you may leave this field blank. The next option is to type in a "Book ID" number. Once this is done, all other details for their book such as ISBN Number, book name and its price will automatically be updated; therefore there is no need for you to input this data. After inputting a "Student ID", the students' details will also be updated including the form details, showing you the form group, ID, room and form tutor's name. The final



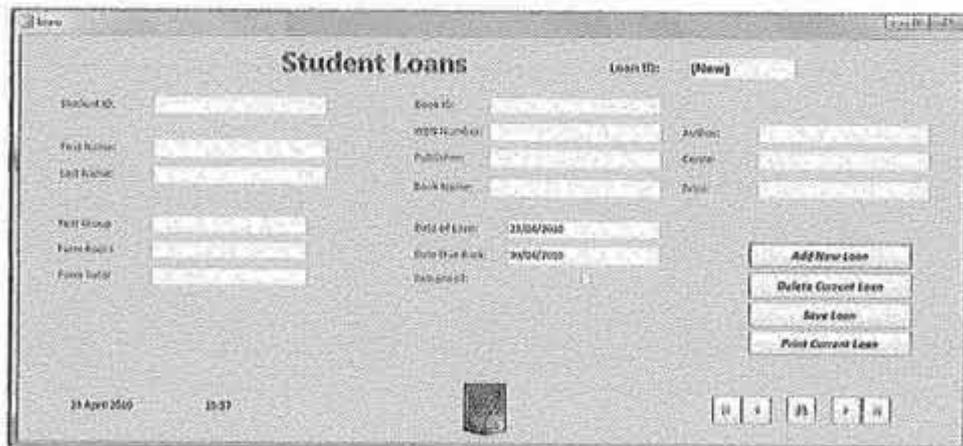
thing for you to do in order to add a loan is to select the date of the loan starting and the date that the loan will be due back. When selecting these dates, a small calendar icon will appear. If you select this you will be shown, as you can see from this screenshot, you can select the date from a calendar making it much easier to use.

Once a book has been returned, you can open this loans table and navigate to the appropriate record by selecting inputting the loan ID. After recalling the details you can then select the option "returned" to show the book has been returned. This will therefore show the loan has finished but will however keep a record of the loan should it be needed in the future, e.g. If the book was found to be damaged.



Once using the database you will wish to add new loans to the system in order to link a particular book to a student with a date the loan was made, therefore making a record of this on

your system. In order to add a new loan, simply select "Add new loan". This will produce a blank page as show below. You can then begin to add data to the record.



Similarly to adding a loan, you may wish to completely remove a loan from the database. In order to do this navigate to the

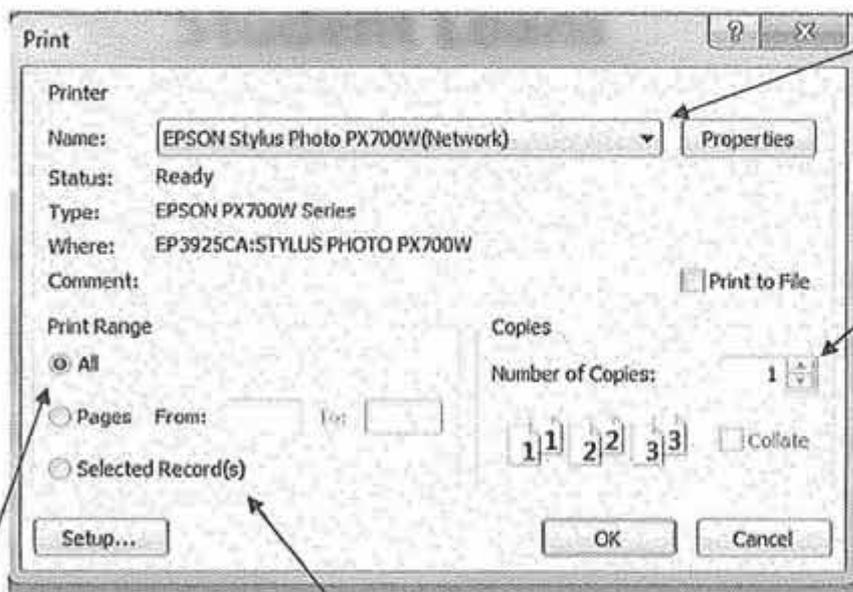
record that you wish to delete and select the option "Delete Current Loan" This will erase this record from the database (This cannot be recovered so ensure that you are on the correct record and are sure that you wish to completely remove the loan)

### **Save Loan**

Another button that is available for use within the data input form is the "save loan" feature. This will save the record that you may have created or one you have just edited details about. It is very important to save the loan after you have made any changes to the database as this will avoid data loss when closing the database or having any problems with the PC you are using.

### **Print Current Loan**

The final button available on the "loans" screen is "print current loan". This button can be used to print the current loan that you are viewing or all loans within the database in the style of the form you are viewing. This is if you wish to display all of this data about only one record. Otherwise, you should use the "reports" option within the database as this will give a much more professional layout to anything printed from the database. Upon selecting this button you will be presented with an options box similar to the one shown below:



Select the printer you wish to use from this drop-down menu.

Enter the amount of copies you wish to have printed.

Select "All" to print all records currently in this part of the database

Select "Selected records" to print the record that you are currently viewing.

## Books Menu

The Book ID is automatically generated. This will be unique for each book and each book should be marked accordingly.

Enter the ISBN number here. The ISBN number should be 15 digits long, in the format:

111-1-11111111-11-1

Enter the Book name here

Enter the author of the Book here

Enter the book publisher here.

The price of the book should be entered here. The Book cannot have a value of £0.00 and the currency will be automatically entered.

Enter the genre/subject of the book here.

You can also select the "Books" option on the main menu in order to add, edit or delete a book from the database. You may wish to do this if details about a book are incorrect, a book has been lost or any new books have been purchased which need to be added to the database. Once again you can use the navigational options that have already been explained in order to navigate between records on the database. This data input form again has the "Save Book" option which must be used each time a new book is added and the "Print book" button should details about a particular book with the be printed. If you wish to add a new book to the system you can select "Add new book". Once all details have been entered, the Book ID will automatically be generated as this is unique to each book within the database. It is also recommended that each book is marked with this Book ID for identification purposes.

The ISBN number must be entered in the correct format for the database to accept it. This is numbers and no letters and the database has been designed to input the – between the ISBN numbers automatically. If you wish to remove a book from the database for example, if it was damaged or disposed of, you can navigate to that record and select the "delete book" option which will permanently be disposed of.

### Save Book

Within the "Books" form, the "save book" option is the first button displayed. This must be used to save any changes made or new records added to the database. It is important to do this before closing the database and after any changes are made as you may otherwise lose data.

### Print Book

This "print book" option can be used to print any records of books that are stored in the database. This means that you are able to print all books within the database or just the record that you are currently viewing. Upon selecting this button a print options page will be displayed:

Select "All" to print all records currently in this part of the database

Select "Selected records" to print the record that you are currently viewing.

Select the printer you wish to use from this drop-down menu.

Enter the amount of copies you wish to have printed.

### Delete Book

This next button can be used in order to remove a book from the database. After pressing this button the Book will be permanently deleted from the website. This means that you will not be able to process any more loans related to the book. Should you receive an error saying "Cannot delete as related records in another table". This means that you cannot delete a book because it is currently being used within another table E.G. the book is currently on loan to a student.

### Add Book

This final button can be used in order to add a new book to the database. The book ID will display (new) until you enter details for the record where the book will then be assigned a unique number. Remember to use the above "Save book" button to save the record before closing the database.

## Students Menu

*The Student ID is automatically generated within this field.*

*Enter the Students First Name and Last names here.*

*Select the arrow and select a Students form from the list. If the Form is not listed, use the "Form groups" form to add a new form group.*

*Any additional notes you wish to add about students can be entered here.*

The screenshot shows a web-based form titled "Students". The form is divided into several sections. On the left side, there are input fields for "Student ID" (containing "10"), "First Name", "Last Name", "Form ID" (with a dropdown arrow), and "Additional Notes". On the right side, there are four buttons: "Save Student", "Print Student", "Delete Student", and "Add Student". Below these buttons, there are three more input fields: "Year Group" (containing "10"), "Form Room" (containing "1AAS"), and "Form Tutor" (containing "Mr Arnold"). At the bottom of the form, there is a date and time display showing "26 April 2010 12:01" and a set of navigation buttons.

*Upon entering the Form ID, the Year Group, Form Room and Form Tutor are automatically shown here. You cannot edit this data here. To do this, use the "form groups" form.*

The students menu is the next option available from the main menu. This form is used to add a new student to the database or to edit/delete students from the database. This is therefore most likely to be used at the start of the year to add new students and delete all of the older students that have now left. The first option on this form is once again to "save student". This will save the record for any new students or student details that have been amended. The next option is to also print all options for any particular student.

The third button on the firm allows you to delete the record that you are currently viewing. This database has relationships between records however, so if a student you are trying to delete currently has a loan, this will not be possible. Once again, to add a new student to the database you can select the option 'Add student' option. You must once again fill in all details in order to add a new student except for the 'Additional notes' section which has been provided to add notes to a student's record, for example if they have a habit of not returning books or damaging them. Upon the Form ID being selected from the drop-down list, the year group, Form room and form tutor information will be automatically displayed.

### Save Student

You can use this button in order to save the records of any students that you have just created or edited. It is important to use this button before closing the form or database as you may otherwise lose any unsaved work.

### Print Student

This button can once again be used in order to print all records or one individual record shown within the "students" part of the database.

Select "All" to print all records currently in this part of the database

Select "Selected records" to print the record that you are currently viewing.

Select the printer you wish to use from this drop-down menu.

Enter the amount of copies you wish to have printed.

### Add Student

You can use the "Add student" button in order to add a new student to the database. After selecting this button you will be presented with a blank form as shown below which you can fill in.

This image shows how the blank form will look. The student ID currently shows (New). Once the data has been entered to create a record, a new Student ID will be automatically created. Remember to use the "Save student" button before exiting in order to save any details that you enter.

### Delete Student

This final button can be used in order to delete a student's details from the database. This means that the student will be completely removed and this cannot be removed. Should you receive an error message saying "the record cannot be deleted as related records currently exist" this means that the student currently has a book and therefore that loaned book needs to be removed first.

## Form Groups Menu

*The Form ID is automatically generated here.*

*Enter the Year group of the form here.*

*Enter the Form Room here. This must be in the correct format E.G. TA45.*

*Enter the form Tutors name here, this must begin with Mr, Mrs, Ms or Miss*

The final data input form used on the database is to enter or edit/delete any form groups for the school. This again has the same option allowing you to save the form group should any changes have been made as well as the option to print a form group should you need a paper copy of the form group.

You also have the option to add a new form and to delete a current form. In order to add a new form all fields must again be input and validation rules exist so that teachers' names must be in the correct format and also room numbers must be in the correct format. The same navigation options are also used on this form in order to move between records to search for a particular record. Due to the relationships in this database between records, you will not be able to delete a form group if it currently has students assigned to it, as this would cause errors within other tables therefore students must be moved to a new form group or removed from the database first.

### ***Save Form Group***

The first button displayed within the form is the "Save form Group" button. This button must be used once any changes are made the database or a new record is created

in order to avoid data loss.

### ***Print Form Group***

The print form group button can be used in order to print all or one of the records for each form group stored within the database. Once this button is selected, the printer dialogue is displayed (As shown on next page) in order to send the records to the printer.

Select "All" to print all records currently in this part of the database

Select "Selected records" to print the record that you are currently viewing.

Select the printer you wish to use from this drop-down menu.

Enter the amount of copies you wish to have printed.

### Delete Form Group

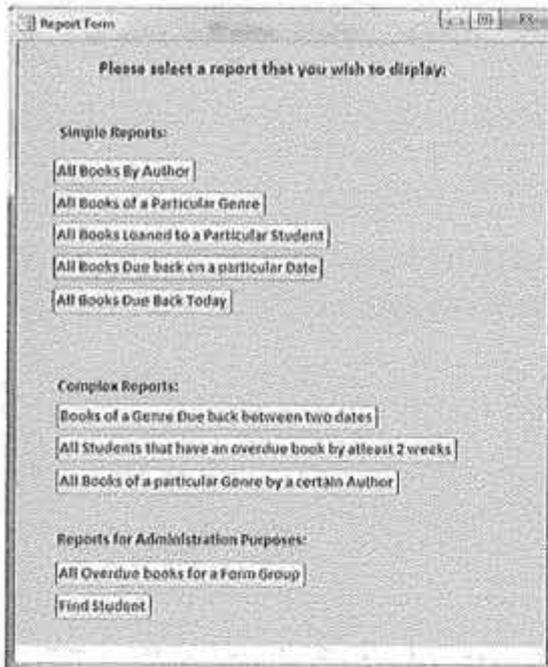
This next button can be used in order to remove a form group from the database. It must be noted, that once the record is removed it cannot be recovered. This is therefore only recommended to be done at the end of each year when details about form groups may change. In order to prevent mistakes from being made, the database will not allow a form group to be removed whilst students are still assigned to the form. The students must therefore be moved to another group or also removed from the database.

### Add Form

The final button displayed on this screen is used in order to add a new form group to the database. Upon selecting this

option a blank form will be shown where the "Form ID" will be automatically generated. Other parts of the database E.G. the "Students" form will be automatically updated to include this new form group.

## Reports Menu



After selecting "reports" from the main menu, this screen is displayed. All reports that the database can produced from the database are displayed here and they are split into three sub-categories. These are "Simple reports" which are ones that air fairly easy to understand, "Complex reports" which require slightly more input and "Administration reports" which you may be requested to do. These reports rely on the system time therefore It is important to ensure the computers time and date are set correctly.

The reports available within this menu are listed below with a short description. See below for instructions on how to input data to run these reports.

**All Books by author:** When launching this report you will be asked to enter an authors' name. After entering the name and selecting "OK", a report will be produced showing all books by that author that are present on the system.

**All Books of a particular Genre:** You may find that at some time you wish to search for a book that is of a particular Genre or subject. After selecting this report you will therefore be asked to enter a Genre that you wish to search for, for example "Mathematics" or "Horror". After selecting "OK" all results for this value are then found.

**All books loaned to a particular student:** This report may be useful if students are only allowed a maximum number of loans or you suspect someone to be taking out too many books. After selecting this button you will be asked to input a "Student ID". After selecting "OK" all books on loan to that student will be displayed.

**All books due back on a particular date:** At some point you may wish to look at books that are due back on a particular date, for example you may wish to know the day before how many books are going to be returned the next day. After selecting this report you can therefore input a date and select "OK". All books due back on the date entered will be displayed. The date should be entered in the format: DD/MM/YYYY E.G 24/05/2010

**All books due back today:** Should you wish to look at the books which are due back into the library today you can select this report. This report is fully automated and does not require you to input any information to the database. A list of all books due in on today's date will be displayed.

**Books of a particular Genre due back between two dates:** This report requires you to input three sets of data. These are the Genre/Subject of the book, the start date you wish to search for and the end date you wish to search for. For example if you want to find a Horror book due back between The 1<sup>st</sup> of March 2010 and the 5<sup>th</sup> of April 2010, you would be entering "Horror", "01/03/2010" and

"05/04/2010". Note that the dates must be in the format DD/MM/YYYY. You would then be displayed all horror books that are due back between these dates.

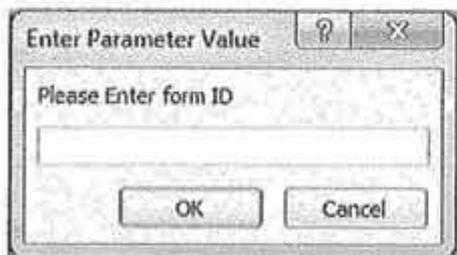
**All students that have an overdue book by at least two weeks:** This report will look for all students that have a book that is at least two weeks overdue. This report requires no user input as the system is already aware of the current date and can therefore calculate two weeks before this. You will therefore be produced a list of books and students which have the books that are more than two weeks overdue. You may wish to use this report to pass on to senior staff in order to get the books returned.

**All books of a particular genre by a certain author:** This report requires two sets of data to be input in order for it to work correctly. This is the Genre/subject, such as "ICT" and the Author that you wish to search for, such as "David Rayner". You may wish to use this report should you find that a student does not know the name of a book but knows what it is about and who the author is. This could then be used to help identify the book that they are looking for.

**All overdue books for a form group:** This report may be one of the most regularly used as it is designed to be sent to all form tutors on a regular basis, e.g. Once a week. This report simply requires you to input the "Form ID" for each form group and will then display all overdue books for students within that form, which their form tutor can then ask them for. This report will automatically list the books below a short heading to inform form tutors of the problem and ask them to resolve it. It also provides a summary of the total cost of books overdue and the total number of books that are overdue for each form.

**Find Student:** You may wish to use this report should a student forget their "Student ID" number or should you wish to look at their details but not know their ID number. Upon running this report you can enter a First name OR Last name with the report showing all results found. Should you wish to search just by last name, leave the first name option empty selecting "OK" and the report will search only for the last name.

When using the above reports you will be asked to enter data in order to run the reports.

A screenshot of a Windows-style dialog box titled "Enter Parameter Value". The dialog box has a standard title bar with a question mark icon and a close button (X). Inside the dialog, there is a text input field with the placeholder text "Please Enter form ID". Below the input field are two buttons: "OK" and "Cancel".

This is an example of one of the questions you may be asked when running a report. In this image, you are asked to "Enter form ID". Should you wish to search for Form ID 10, simply type "10" into the box, selecting the "OK" button. Should you wish to stop running this report select "Cancel" and you will be returned to the "reports" screen.



## Backing up the database

It is highly recommended that a backup of the database is done on a daily basis, therefore should anything go wrong with the database, at worst, only 1 days worth of records will be lost. In order to do this it is also recommended that the database is backed up to an external drive in case of a computer hard disk failure.

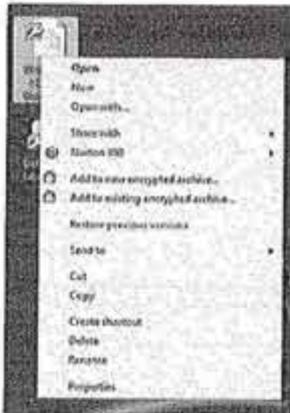
There are two main ways to back up the database with the first one being the most ideal option. If this is not possible, however, it is possible to manually back up the database by making a copy of the database file once a day.



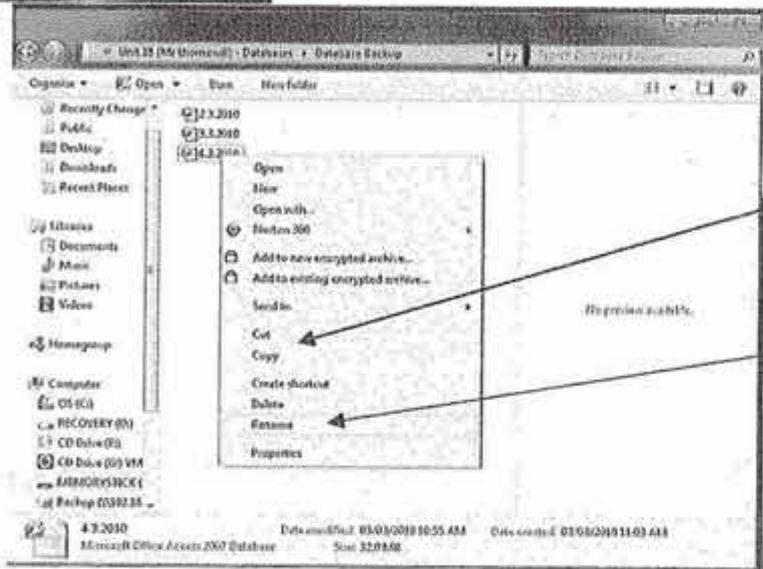
The first method to backup the database can be done by selecting the Microsoft office Button in the top left corner of the software as shown within this image. After doing this a menu will then open where you can select "Manage" on the left side of the menu. The right side of the menu will then produce three options with the second one of these saying "Back up database". After doing this, a box will open asking you where you wish to save the database. After navigating to where you wish to save the database file you then

simply select the "save file" and a copy of the database will then be made for you. The name of the file will automatically have the date entered at the end therefore automatically keeping backups logically organised for you.

The second method to back up the database is by doing it manually:



To do this simply right select the database file and select 'Copy'. Navigate to a file on a removable drive, right select again and select the option 'Paste'. It is also recommended that you rename the database to the date that it was backed up so that they can be kept in a logical order.



Option to 'Copy' or 'Paste' database file.

Option to 'Rename' database file.

It is recommended that regardless of which backup method you use, that backups are all stored within one single folder named "Database backups" and a new folder is created for each year, therefore ensuring that backups can be easily found in the future should they be required.

## Acceptable data and Validation used within the database

This section of your user guide shows the validation that is used within the database in order to stop incorrect data being entered. This will therefore show the validation used and the acceptable data that can be put into areas of the database which are restricted.

### Student Records:

**Student ID:** This value cannot be changed and is automatically generated by the database.

**First Name:** This cannot be longer than 20 letters and must be letters only – the database will not accept numbers in this field.

**Last Name:** This cannot be longer than 20 letters and must be letters only – the database will not accept numbers in this field.

**Additional Notes:** This can contain any combination of letters and numbers. The limit is 200 characters including spaces.

### Books records:

**Book ID:** This value cannot be changed and is automatically generated by the database.

**ISBN Number:** This field can be numbers only and must be in the format: 123/1/12345678/12/1

**Book Name:** The book name can be any text or numbers up to 30 characters.

**Author Name:** The Author name can be text or numbers up to a limit of 20 characters.

**Publisher:** The publishers name can be any text or numbers up to 20 characters.

**Genre:** The Genre can be any text or numbers up to 20 characters.

**Price:** The price cannot be £0.00 and must be in the format 0.00. The currency will be automatically added.

### Form Groups Records:

**Form ID:** This value cannot be changed and is automatically generated by the database.

**Year Group:** The year group can only be numbers and must be between Years 7 and 13 (inclusive).

**Form Room:** The form room must begin with "TA", "TB" or "TC" and then followed by a two-digit room number E.G. TA25.

**Form Tutor:** The form tutors name must begin with Mr, Ms, Miss or Mrs and has a limit of 20 characters.

**Loans Records:**

**Loan ID:** This value cannot be changed and is automatically generated by the database.

**Date of loan:** The date of the loan must be entered in the format DD/MM/YYYY or entered using the date picker.

**Date due back:** The date due back must be entered in the format DD/MM/YYYY or entered using the date picker. As the minimum loan length must be one week, this date must be at least 7 days after the today's date.

**Returned:** No validation used, select the check box to enter a "tick" to confirm the book has been returned back to the library.

## FAQ (Frequently Asked Questions)

### 1. Can multiple users access the database at once?

Yes, it is possible for multiple users to work on the same database file and the same time. It must be noted however that if different changes are made to the same piece of data and multiple copies are saved, changes may be lost; therefore it is important to save the database regularly.

### 2. Why won't my report print?

Your printer may not be installed or working correctly. After going to the "Print" menu, you should see your printer listed. (Make sure it is selected from the drop-down list). If the printer is listed it is installed correctly and the error is caused by a communication failure between the PC and printer or a Printer fault. If your printer is not listed it needs to be installed from the software provided with the printer. If you are able to print from elsewhere other than the database, re-install Microsoft Access. If this does not fix the error, please contact our Technical Support team on the number provided. (This is available from the main menu, selecting the "About" menu.

### 3. Why will reports not display data after entering a date?

Reports need the date entering in the correct format. This should be in the format DD/MM/YYYY e.g. 21/02/2009

### 4. Why does the time not update on the main menu?

This is the way that the database has been designed and is not a fault – it will show the time the database was last refreshed when it has been interacted with. After selecting any buttons or menus the page should refresh and the new time will display.

### 5. I'm running Microsoft Office 2003, why will the database not open saying that it is "Not compatible"

Features on this database are created with and only compatible with Microsoft Access 2007, therefore you need to upgrade to this software (or newer) to open the database.

### 6. Why does the database keep opening and displaying a "Macro" error?

This is a security feature as Macros used on the database could potentially be dangerous. The macros used within this database are safe. If this error is displayed, simply select the error and select "Run all macros". To stop this error in the future you may go to the "Access Options" available from the "office logo" in the top left corner of the screen. In here, you can choose to permanently enable all macros.

### 7. What is the best way to close the database?

To close the database, ensure that all changes have been saved and then return to the main menu. Use the "Exit" button on here to close the database and software. Avoid using the normal "X" button in the corner of the screen as this may cause data loss if you have not saved.

**8. Why can I not delete a record and keep getting an error message about related records in other tables?**

This error message is produced if you try and delete a record that is currently being used in another record, such as trying to delete a book from the system when the "loans" table currently shows a book is on loan to a student. This is to protect the database from accidental deletions being made. In order to delete the record, first delete any related loans or other records that are within the database.

## **Database Evaluation**

I am now going to complete the evaluation for my Database project as a solution that was made for a library. At the start of this project I was informed that the school currently uses a flat-file database system within Microsoft Excel where a new row is added for each loan made. After looking at the clear disadvantages of this system, not limited to the fact that the database will have lots of repeated entries, errors are very likely and that it takes a long time to find a particular loan, It was decided that I would design a new database system for the school to use.

### **Database evaluation**

After looking in-depth at the current system such as what information it holds and ways that it could be improved I then looked at the aims that would be suitable for the new system. The first aim for the new system was that specific records should be able to be accessed easily, this will allow the users of the system to find a specific record should it be overdue, there be any questions about a loan made or should the book be returned. My system meets this aim very well as there are a number of ways that an individual loan can be found. This can be done by the use of a number of queries such as "all overdue books by a particular form". If however the user wants to access one specific record they can simply navigate to the "loans" data input form, select the "search" button and type in the Loan ID, the system will then display details about this loan. Alternatively within the data input form the user can use the "previous" and "next" button to navigate through all of the loans made, clearly displaying the details of each one. Evidence that this was working was done during the testing stage to ensure the buttons and search facility were working. This feature within the database therefore clearly meets this aim.

The second aim of the database was that you can see how many books a student currently has out on loan. The reason behind having this as an aim of the system is that librarians can look how many loans students have if they have a limited number of loans and can produce a report if a student has lots of overdue books which they need to make senior members of staff aware of. My system meets this requirement as a user can navigate from the "Main Menu" to the "Reports" button. This will then produce another menu where the report "All books on loan to a particular student" can be selected. Once selected the user can then input a Student ID number into the database and any loans that this student has will be displayed. My system therefore meets this aim well and allows users to be able to find out this information very easily meaning this part of the database will be ideal for their needs.

The third aim that I set out in my analysis of the current system is that the new system must have a "comments" or "notes" section that allows the user of the system to enter any other information about a particular student. They may wish to do this as they may find particular students regularly damage or lose books and therefore may wish to make notes of this on here along with any charges they may make if the book is not going to return. My system meets this aim well as I have included a notes section with no validation rules that allows the user to enter information about particular students. This is displayed as a text entry box within the data input form for "students". This means that this feature can easily be used by opening the relevant student's record and simply typing in any required notes for that student.

The fourth aim of the system that I have developed is that it can be used by multiple users at the same time. This means that during busier times within the school such as breaks or lunchtimes, it is possible for multiple users to make or edit loans within the database at the same time. A strength of this is therefore that people will not have to wait as long when there is a higher demand for loaned books. One disadvantage of the way this system works is, however, that there is a possibility for one user to lose data when multiple users are working on the same database. This is because the database will not let users edit different things on exactly the same record at exactly the same time, possibly causing one user to lose the changes they made when saving. This is however only a very minor weakness of the system as it is unlikely that the users of the system are going to be accessing the same record at the same time, and therefore they should not be affected by this.

The fifth aim of the new system was that there should be no repeated attributes within tables meaning data is not un-necessarily repeated. This aim has clearly been met by my system as I have designed four tables which are all relational to each other. This can be seen within the designs for my database where I have completed the "normalisation" process to ensure that no attributes are repeated. This should save the user's time, reduce errors and therefore increase the reliability of the system.

The next aim of the system was that queries can be run on the database to find specific information and these can be produced as professional-looking reports that can be printed from the system. I think that my system also meets this requirement well as this is one of the strongest parts of my system; this is because the queries are easy for the user to input data with a dialogue box and message being displayed when appropriate. Once the data has been input by the user the query can be run and results are displayed on

appropriately designed reports which can be printed from the system.

The final requirement for the new system was that I needed to consider how complicated the database will be and things such as training for the users so that they know how to use it appropriately. I have therefore met this requirement for my system by producing a user guide that shows any users how they can use the system effectively. It also contains information such as a "frequently asked questions" section so that basic user-errors can be discussed. I think that this has worked very well as it means that people can read this user guide and follow it through with the system on their screen, making a logical progression along all parts of the system.

I am now going to look at the feedback that was received for five questions that were asked a potential user of the system to look if it met their expectations and requirements.

**1. Does the database store all of the information you expect in a relevant format?**

*"The database stores all of the information that I'd have expected and does this in a really easy to understand way. I can use the menus to find my way around and edit the data for each form group, student and book really easily. I like the way that all this information is fetched for me automatically on the loans menu so that I don't have to keep re-typing it like the old system! It also stops me making as many mistakes like I did in the old system by entering wrong information. The only data that I think may be missing is contact information for the student such as Address and Telephone numbers of Parents/Guardians; however this is currently available on a separate system that we use."*

**2. Do you find the interface provided easy to use and navigate?**

*"Yes, the programme that I am using and all the menus let me really easy find my way around – there is only around 5 different screens which I can see information on and all information is really clearly given. I can also click a button to run reports and preview the results before printing them. The only part of the interface that I do not like is that on the 'reports' menu, where you have to select the various reports from the buttons available. I think that this could be made better by making the buttons equal size and by spacing the reports out more – they currently look a bit confusing until you read each individual one".*

### **3. Does the provided Manual give enough detail whilst being easy to understand?**

*"The manual that has been provided is really useful but has to be read carefully if you're not technically minded like me, but once I had read it slowly I was able to use it alongside the system for a few days before being able to use it by myself, so far I've not come across anything that I can't find how to do in the manual. The FAQ is useful as this has provided the answers to a few questions I have found, I would however like a glossary of Terms used such as "Macro" as I was not sure what this was, Despite this I was still able however to follow the instructions and the error with macros that I had was resolved."*

### **4. Do you think the reports that can be produced are relevant and useful?**

*"The reports are really useful and I have begun using them already. Some I do not think are so relevant such as the one "All books due back today" as I may not use this very often. I do however think that some other reports such as "all overdue books for a particular form" work really well and are going to save me hours of work. Previously I had to look through every record to see if a book was overdue yet and then write down each of these on a piece of paper for that form. I then had to find a form tutor's name and form room, write the letter and the cost of each book, I had to do this every week! Now all I need to do is type in each for ID and if there are any results send the report to the printer. This automatically contains all of the information I used to calculate and more such as the total amount overdue and the total cost of the books, it's really useful."*

### **5. Overall, what do you think of the new Library database that has been designed?**

*"I really like the new library database system, it is saving a lot of time when adding or changing records of loans and it will be much faster when new students come to the school or we make a bulk purchase of books. It did take a short while to get used to when compared to the old system, however I think that it is a worthwhile improvement that makes my job much easier. I have also recently found error messages appearing when I have entered information wrong and, although I am unsure how this works, it has stopped me entering the wrong data a few times already so that is also very useful."*

#### **Analysis of Feedback**

I have now asked a number of questions to a librarian of our school who will potentially be using this new database system in order to

gain their feedback and look how the product could be improved in the future. The first question that I have asked for feedback was to find out if the database is storing all of the information that the user will require. After asking the user this question they appear to like the new database and have already noticed that it means they do not have to keep entering the same data numerous times. From this first question feedback has been provided however of how the database could be improved in future. I would therefore look at addressing the problem that no student details such as Address and Telephone numbers of Parents/Guardians are available. I would therefore look in the future at implementing these details and also examine the other system that the Librarian currently uses for details such as this. I could then look at making a further revision to the database to include this system, therefore effectively merging two systems into one. This would then mean librarians do not have to keep moving from one system to another and would be able to work more efficiently.

The second question asks the user how easy they find the interface to use. The feedback received from this question was good saying that the interface is easy to use and easy to navigate around, the user is clearly happy with the interface provided for them to use. It was suggested however that the reports menu is a little confusing and the buttons could all be made the same size. After reviewing this menu I think that future improvements could be made in order to address this issue. I would therefore look at increasing the button and text size to fit over two lines whilst also cutting down the amount of text in some buttons. I would also look at increasing the size of the whole menu and splitting the reports into a more logical order to make them easier to find.

The third question that I have asked the librarian was to find out how useful the Manual provided was. The feedback received for this question was that the user guide is detailed and contains all of the information required in order to use the database effectively. The one improvement that was suggested from my user feedback was that they did not understand some of the technical terms as they are not technically-minded. I would therefore in the future look at implementing a glossary page which looks at definitions and explanations of any technical terms, as this would allow the user to gain extra clarity on any points that wish to find out more about or any technical terms that they do not understand.

The fourth question asked was to find out about the reports that were available, if the information was accurate and if the provided reports were useful. The feedback given from the user is that some of the reports are very useful such as "Overdue books for a particular form" as they are going to save a lot of time, when compared to the old system where they had to be manually collated

and the report created. This therefore shows that the user will find some of the main features of the database useful and it is meeting its intended purpose. In order to address the issue of how useful these reports are, I would look at changing these as an on-going improvement once the database has been implemented. Further feedback could then be gained about any reports that have not been used and any others that are still required. These could then be added to the system in the future.

The final question asked was to gain an overall conclusion of how useful the whole database will be to the librarians. The response received from this was very positive saying that it will make their jobs much easier and save a lot of time, therefore meaning the database is meeting its purpose by helping to keep loaned books much better managed. The improvements that could be made in the future to address the above issues should be part of continuous development which could be used with the database where continual feedback could still be received so that the database could be developed further when required. My database has therefore clearly pleased the end users who are going to be using it and they clearly feel that it is going to benefit their job compared to the old system.

I am now going to look at the strengths and weaknesses about my database in order to look at what has worked well and what has not worked so well. With these weaknesses in mind, I will then look at possible improvements that I would choose to make to the database in the future.

I think that the first strength of my database is simply the way that tables have been designed so that no data has to be repeated and all fields that are required have to have data input before the user can save the record. This means that the user cannot input incomplete records onto the system as this may cause errors at a later date. Another strength of the way this database has been designed is the relationships between tables and how this allows subforms to work. This will be useful to users of the database as they can enter a foreign key into one data input form and the subform will look for this primary key in another table, fetch the record and display all necessary data in a clear way. This means the users of the database should have all information that they may require on each form. Another strength of this database that is linked to the relationships between tables is the referential integrity that I have used. This means that a record cannot be removed if it has related records within another table, for example a Form Group cannot be deleted if Students still exist within it. This is a good feature of the database as it stops records being accidentally or deliberately deleted which could cause errors within other records and tables.

I also think that another strength of the final database I have created are the validation rules that I have used, this has also come from feedback from one of the end users who has been using the database for a few days and already found the validation rules have prevented her from entering incorrect data. These validation rules therefore work well, enforcing the reliability and accuracy of data within the database. Finally I also think that the fluent interface and clear menu buttons I have created are a strength of the design. This is the end user cannot see or access the tables which the database runs on as this could confuse them and also open the database up where changes may be accidentally made. By providing the data input forms and menu as an interface that the end users can use, I am protecting the database from the end users as this will hide areas that they do not need to access. The clear buttons also mean that as well as being able to easy see data on the forms provided, navigation from form to form or between records as the buttons are fairly large and simple to understand.

Despite my database being a success and my end user clearly finding it useful, there are however a number of weaknesses in the design that I have found. The first one of these was noticed during the testing of my Form designs. Although I have included a macro to refresh the time and date every second, the time and date do not appear to update on the input forms unless an option is selected to refresh the screen (such as navigating to the next record). This is however working on the main menu with the time being constantly updated. Another weakness is that the end-user who has been using the database for a few days said that despite having the manual, the database still took a little while to get used to, this is a weakness as the users will therefore take a while before the database becomes fully functional after they learn all features available within it.

Other weaknesses about the final system include the logo that has been used. This image is fairly low-resolution and is on a white background. This means when the logo is placed on other backgrounds such as Green, a white box can be seen around the image. Whilst this does not affect the operability of the database it does mean the forms displayed do not look as professional as possible. I do however think that the main weakness to the database is that table have no awareness if a book is currently on loan and there is no way of cross-referencing this. For example, if a loan were to be made and a book ID incorrectly entered for a book that is already on loan, the database would not be aware that this is already on loan. This means no error message would appear and the records would show the same book being on loan twice at the same time. The final weakness of this system is from a security point-of-view. This is that the database does not require a password to be accessed and therefore should it be left unattended,

unauthorised changes could be made. Whilst PC's can have user passwords set, this means if a user is already logged on, anyone could have access to the database. It is therefore currently important to close the database and log out when leaving the computer.

Taking these weaknesses into account, there are therefore a few improvements that I think could be made to the database. Firstly I think the most major change would be to add on the books table an "On loan" Boolean field that is linked to the loans table. This means if a book is in the loans table AND the value "returned" is false, the "on loan" option is selected, therefore showing that the book is currently out of the library. This could then be used within reports to increase the accuracy of results. I also think that the data input forms could be changed to force the Date/time code to refresh every second so that the time and date are continuously correct, regardless of when that page was accessed. Finally I think that the last improvement could be made to aid users when learning how to use the database. I think that the best way to do this in the future would be to develop animated tutorials showing the user how to operate each element of the database. After doing this, they should also be able to practice each element once at a time before moving onto the next stage. Having interactive tutorials rather than a manual will provide a more hand-on experience and end users will learn faster by doing this.

### **Performance evaluation**

I am now going to look at my performance during this project, the strengths and weaknesses of my performance and how I could improve my performance for projects such as this in the future.

I think overall that I have performed very well for this project. I firstly started by producing an analysis of the current system and how a new system could correct the problems that were found from using the old system within Microsoft Excel. I think that my performance during this section was very good as I critically analysed the current solution and the way that a new solution could meet these problems. The only weakness to my performance during this section was the lack of information I was given to analyse. This was because I was simply told the type of database currently used and given an example of a few entries into the database. My performance on this section could therefore have been improved by being given more detailed information about the way that the database was used and things that the users currently missed from this type of database, as I would then have been able to do a more detailed analysis.

After the analysis stage I then moved on to begin designing the solution for the problem I had been provided with. A strength of my

performance during this section was the effort and dedication that I put in, in order to get the designs and plans finished to a high quality in a reasonable amount of time. I think another strength of my for the planning that I have completed was that I was very well organised and that I followed a logical progression through the design stage processes in order to ensure that the project was accurate and all relevant stages were completed. A weakness of this design stage was however the time that the designs took especially when completing the normalisation process. As I felt that the normalisation process was something I was more unsure of, I spent some time researching the process and then took my time when normalising the data as this is a very important part. This is because if I were to not fully normalise all of the data within my database, repeated data would be likely to exist and my database would become less reliable as a result of this. The weakness of this part was therefore the time that it took to research this stage and then carry it out however this was well worth the time that was invested.

After planning the database I then moved onto the implementation stage. I think that the strength to my performance was that I had previous knowledge of databases and therefore actually setting up this database I did not find too difficult. I also think that I showed good dedication to the project putting in a lot of extra hours than expected due to the amount of time that the forms, queries and reports took to set up correctly. A weakness however of my implementation stage is that the database was designed using software only available to me for a limited number of hours per week, this meant that the amount of time I could actually spend implementing the project per week was limited. Whilst this meant that had to ensure I was well organised to use my time effectively. I feel that this stage could have been improved should I have been able to use this system for a longer amount of time, therefore allowing me to implement the system much faster and much more effectively.

The testing stage was then completed after the implementation stage had been completed. I think that a strength of this stage was the commitment that I put into the project in order to be able to finish the testing stage as this was a very large part of the project. I showed my commitment by firstly completing a testing plan of over 100 tests to ensure the database was working correctly; I then carried out all of these tests making revisions to the database where necessary once faults were found. All of this was done before moving on as I was committed getting the database fully functional and running in the shortest amount of time possible. Once again, organisation was also a key skill of mine used here as I planned out all testing to be done before carrying it out as this was the most time-effective method available.

I therefore think that my performance put into this project as a whole was very good and that I have shown a good amount of effort and commitment in order to get the database designed, created and tested within a reasonable period of time. Summarising from above, I do however think that in order to improve this further I could have allowed more time for access to resources that were required to implement the project, and that longer time spent on a more in-depth analysis at the start of the project may have resulted in a more suitable system as more of the user's requirements may have been found. I do however think that overall my performance can be clearly seen from the final database that I have produced as this is of a high quality and meets the needs of the users that I identified at the start of the project.