

Edexcel GCE

Applied Information and Communication Technology

Unit 7: Using Database Software

January 2012

Scenario

Paper Reference

6957/01

The scenario should be distributed to candidates at least three working weeks before the examination.

Note: There are **no** practice files for this examination.

This scenario should be used for the purposes of preparing candidates for the examination. This copy **must not** be taken into the examination. The information contained in the scenario will be included in the examination paper.

Further details are in the Instructions of the Conduct of Examinations (ICE), available from the Edexcel website for this qualification.

Edexcel will not accept any request for special consideration should candidates be given the incorrect scenario for the examination they are sitting.

Turn over

W40717A

©2012 Edexcel Limited.

1/



edexcel 
advancing learning, changing lives

Scenario

Berrymill Secondary School Trip

Charles Howells is a science teacher at Berrymill Secondary School. He is arranging a four-day trip to Paris for three teachers and 20 students. The dates for the trip will be 14th to 17th March 2012.

Charles needs to collect the students' personal information, including their passport numbers and emergency contact details.

The cost of the trip is £400 per student. The students can make one payment of £400 or spread the cost with four £100 instalments.

Charles has used the Internet to research places of interest, and has decided that the students will visit the Eiffel Tower and the Louvre. The students will be divided into two groups of 10 for these visits. On the 15th one group will visit the Eiffel Tower and the other will visit the Louvre. On the 16th each group will visit the other place of interest.

Charles knows you are studying ICT and thinks you could develop an IT-based solution to help him manage the trip. In the first instance it has been agreed that you will design and build a **prototype** relational database system that will focus on four tasks. The final system will be based on your evaluation of the prototype and your recommendations for further functionality.

These are the tasks.

1 **Sign Up**, which involves:

- ensuring a passport number is present
- ensuring emergency contact details are present
- the generation of a unique student ID, which combines letters from the student's name with a sequential number that is always one more than the highest currently in use
- storing the student details if all requirements are met
- providing a suitable message for Charles so that he knows what is happening.

2 Tracking Payments, which involves:

- selecting a student
- displaying the total amount already paid
- entering the payment amount
- displaying the new balance
- the generation of a unique sequential payment ID, which is always one more than the highest currently in use
- storing the payment details
- providing a suitable message for Charles so that he knows what is happening.

3 Assigning Visit Choices, which involves:

- selecting a visit
- determining whether places are available
- selecting a student who wishes to be added to the list if there are places available
- storing visit choices
- providing a suitable message for Charles so that he knows what is happening.

4 Printing Reminders – A printout of a reminder for those students who have not fully paid including the date, student details, the total amount paid already and the outstanding balance.

The printout will need to look professional and be easy to read.

Edexcel GCE

Applied Information and Communication Technology Unit 7: Using Database Software

9 – 27 January 2012

Assessment Window: 3 weeks

Time: 10 hours

Paper Reference

6957/01

You must have:

Cover sheet, short treasury tag, Students_exam.txt

Instructions

- Complete your candidate details on the cover sheet provided.
- All printouts must contain your name, candidate number, centre number and activity number.
- At the end of the examination:
 - All printouts should be placed in the correct order.
 - Use a treasury tag to attach your printouts (**as shown**) to Page 2 of the cover sheet.

Information

- The total mark for this paper is **90**. There are **six** activities in this examination totalling 88 marks. 2 further marks are allocated to Standard Ways of Working.
- The marks for **each** question, within an activity, are shown in brackets
 - use this as a guide as to how much time to spend on each question.
- Use relational database software to carry out the database activities in this examination.
- Questions labelled with an **asterisk (*)** are ones where the quality of your written communication will be assessed
 - you should take particular care on these questions with your spelling, punctuation and grammar, as well as the clarity of expression.

Advice

- Read through the Scenario carefully.
- Work through the activities in order.
- Attempt **ALL** activities.
- Label your printouts clearly as instructed.
- Printing must be undertaken within the examination time.

Turn over ►

P40208RA

©2012 Pearson Education Ltd.

4/1/1/1/1/1/1/4/3



PEARSON

Scenario

Berrymill Secondary School Trip

Charles Howells is a science teacher at Berrymill Secondary School. He is arranging a four-day trip to Paris for three teachers and 20 students. The dates for the trip will be 14th to 17th March 2012.

Charles needs to collect the students' personal information, including their passport numbers and emergency contact details.

The cost of the trip is £400 per student. The students can make one payment of £400 or spread the cost with four £100 instalments.

Charles has used the Internet to research places of interest, and has decided that the students will visit the Eiffel Tower and the Louvre. The students will be divided into two groups of 10 for these visits. On the 15th one group will visit the Eiffel Tower and the other will visit the Louvre. On the 16th each group will visit the other place of interest.

Charles knows you are studying ICT and thinks you could develop an IT-based solution to help him manage the trip. In the first instance it has been agreed that you will design and build a **prototype** relational database system that will focus on four tasks. The final system will be based on your evaluation of the prototype and your recommendations for further functionality.

These are the tasks.

1 **Sign Up**, which involves:

- ensuring a passport number is present
- ensuring emergency contact details are present
- the generation of a unique student ID, which combines letters from the student's name with a sequential number that is always one more than the highest currently in use
- storing the student details if all requirements are met
- providing a suitable message for Charles so that he knows what is happening.

2 Tracking Payments, which involves:

- selecting a student
- displaying the total amount already paid
- entering the payment amount
- displaying the new balance
- the generation of a unique sequential payment ID, which is always one more than the highest currently in use
- storing the payment details
- providing a suitable message for Charles so that he knows what is happening.

3 Assigning Visit Choices, which involves:

- selecting a visit
- determining whether places are available
- selecting a student who wishes to be added to the list if there are places available
- storing visit choices
- providing a suitable message for Charles so that he knows what is happening.

4 Printing Reminders – A printout of a reminder for those students who have not fully paid including the date, student details, the total amount paid already and the outstanding balance.

The printout will need to look professional and be easy to read.

Instructions to Candidates

All word processed documents **MUST** have a header and a footer. The header must contain the activity number. The footer must contain your name, candidate number and centre number.

Minimum font size of 10 should be used throughout.

Screen prints should be large enough to be easily read.

All database reports must have the activity number, your name, candidate number and centre number in the page header.

Activity 1 – Understanding the situation (suggested time 30 mins)

Relational databases need data before they can be used. Data can come from many different sources. Data required by your prototype system is mentioned throughout the scenario.

(a) Use word processing software to create a copy of this table.

Data	Sources			
	Internet	Teacher	Student	System Generated
Student personal details, eg name, address, passport number				
Places that will be visited				
StudentID				
PaymentID				
Outstanding balance				
Total amount already paid				

Complete the table by putting a cross in the appropriate cell to show the original source of the data.

(6)

(b) Use word processing software to answer this question.

The scenario discusses the Sign Up process mentioning that a unique student ID needs to be generated. Within the Sign Up process there is another possibility for uniquely identifying a student. Name the other possibility.

(1)

Evidence to be submitted for Activity 1.

On **one** side of A4, a printout of your completed table and your answer for part (b).

(Total for Activity 1 = 7 marks)

Activity 2 – Structure (suggested time 2 hours)

You will need to use the data file **Students_exam.txt**

This is provided in your examination area.

Study the data file.

(a) Create an efficient database structure that minimises data duplication.

Screenprint the relationships in your database.

(7)

(b) Use the correct data types and key fields.

Produce screenshots in **DESIGN** view of each of your tables showing **only** the field names, data types and primary keys.

(3)

(c) An efficient database must include suitable validation.

Note: you can use the same field more than once if appropriate. In (i)-(v) you **MUST** name the type of validation used.

(i) Screenprint in **DESIGN** view **ONE** example of a **Format Check** on an appropriate field. Ensure you can clearly see the field it is applied to and the format.

(1)

(ii) Screenprint in **DESIGN** view **ONE** example of a **Range Check** on an appropriate field. Ensure you can clearly see the field it is applied to and the range specified.

(1)

(iii) Screenprint in **DESIGN** view **ONE** example of a **Presence Check** on an appropriate field. Ensure you can clearly see the field it is applied to.

(1)

(iv) Screenprint in **DESIGN** view **ONE** example of a **List Check** or **Table Lookup** on an appropriate field. Ensure you can clearly see the field it is applied to and the list items or row source.

(1)

(v) Screenprint in **DESIGN** view **ONE** example of a **Default Value** on an appropriate field. Ensure you can clearly see the field it is applied to and the default value.

(1)

(vi) Screenprint in **DESIGN** view **ONE** example of an appropriate field size. Ensure you can clearly see the field it is applied to and the field size.

(1)

(d) Import the data from the text file provided into your database.

Screenprint each table showing at **least five records**, or all records if there are fewer than five, and the **full record count**. (If the number of fields is too big to fit on the width of the page, all the fields do not have to be displayed.)

(4)

You must assemble your screen prints in the order you were asked to complete them.

(Total for Activity 2 = 20 marks)

Activity 3 – Dealing with sign up, tracking payments and visit choices (suggested time 4 hours)

(a) A sign up form is needed so that Charles can enter and save student details.

(i) Create a data entry form for Charles to use.

For each student the form should collect:

- student's name
- address
- date of birth
- passport number
- parental consent
- emergency contact details.

(1)

(ii) Generate the StudentID and display it on the form. A StudentID consists of the first three letters of their surname and a sequential number one higher than the last number.

Screenprint the form in **DESIGN** view.

Ensure you show how the StudentID has been generated.

It does not need any annotations.

(5)

(iii) Customise the form to make it easier to use.

Screenprint the form in **FORM** view.

It does not need any annotations.

(2)

(b) An automated method of saving a new student record is required.

Create an automated method of saving a student's details.

(i) The automated method of saving should include:

- a message to indicate that the record has been saved
- the form cleared, ready for the next student's details.

Screenprint in **DESIGN** view any macros, code and/or queries you have used.

Ensure that the detail can be seen in full.

(3)

(ii) The process must start by clicking a save button.

Screenprint the form showing the save button.

Screenprint the **OnClick** event properties, or equivalent, of the save button.

(2)

(c) A form is needed so that Charles can add details of payments.

(i) Create a query that will identify the students who have not fully paid for their trip.

Screenprint the query in **DESIGN** view.

It does not need any annotations.

Ensure that the detail can be seen in full.

(1)

(ii) Create a data entry form for Charles to use to add payments.

Charles should be able to:

- select the StudentID from a dropdown box
- see the total amount already paid
- enter the payment amount.

The StudentID dropdown box should use the query created in part (i) as its source.

Screenprint the properties of the StudentID dropdown box. It must be clear that it uses the query created in part (i).

Ensure that the detail can be seen in full.

(4)

(iii) Generate the new PaymentID, which is always one number higher than the last, and display this on the form.

(1)

(iv) Generate and display the new balance.

Ensure that the display is fit for purpose.

(1)

Screenprint the form in **DESIGN** view.

It does not need any annotations.

Ensure that the detail can be seen in full.

Screenprint the form in **FORM** view.

It does not need any annotations.

Screenprint any **QUERIES** or equivalent used.

They do not need any annotations.

(d) Storing the payment details and updating the balance owing is one of the required tasks.

Create an automated method of doing this.

(i) The process must start by clicking a save button.

Screenprint the form showing the save button.

Screenprint the **OnClick** event properties, or equivalent, of the save button.

(2)

(ii) The automated method of storing and updating should include:

- saving the payment details
- updating the FullyPaid field, if appropriate
- displaying a message to Charles when the save has been carried out.

Screenprint in **DESIGN** view any macros, code and/or queries you have used.

Ensure that the detail can be seen in full.

(2)

(e) A form is needed that allows Charles to record visit choices. A maximum of ten students per visit is allowed.

(i) Create a visit choices form that will allow Charles to select or enter:

- the place of visit: Eiffel Tower Group 1, Eiffel Tower Group 2, The Louvre Group 1 or The Louvre Group 2
- StudentID

Screenprint the form in **DESIGN** view.

It does not need any annotations.

(1)

(ii) An automated method of checking and saving the information is required.

The automated method should include:

- a message if ten students have already chosen the visit
- saving the relevant information from the form **only** if places are available
- a message if the information has been saved.

Screenprint in **DESIGN** view any macros, code and/or queries you have used.

Ensure that the detail can be seen in full.

(3)

(Total for Activity 3 = 28 marks)

Activity 4 – Testing (suggested time 1 hour)

(a) Use the sign up form you created in Activity 3 to sign up this student.

Robert Packard
123 The Grove
Berrymill
County Durham
BB3 4EE
DOB: 01/08/1997
Passport number: XY1234567
Emergency contact: Juliet Packard
Relationship to student: Mother
Emergency number: 3131 4444445
Parental consent: Yes

(i) Enter the data into the sign up form and screenprint in **FORM** view.

(1)

(ii) Save the student details and produce:

- a screen print of the message that appears when the student details have been saved
- a screen print of the new record in the relevant table.

(2)

(b) Enter the details of this student on the sign up form:

Waqar Mendi
44 Stockley Avenue
Berrymill
County Durham
BB4 4EP
DOB: 15/06/1997
Passport number: UW2356789
Emergency number: 3131 5678908

Click the Save button and screenprint any message.

(1)

(c) Testing payments for Jane Austin, StudentID: AUS1113

(i) Produce a screen print of the relevant table showing fields for this student.

(1)

(ii) Using the form you created in Activity 3, enter and save a payment of £100 for this student.

Produce:

- a screen print of the completed form in **FORM** view
- a screen print of the relevant table showing the new payment of £100
- a screen print of any updates that have taken place.

(4)

(d) Enter the details of this payment on the payment form.

StudentID: POP1128
Student: Jason Popal
Payment amount: £150

Click the Save button and screenprint the message.

(1)

(e) Use the form you created in Activity 3 to enter and save this choice of visit for Sam Swann, StudentID: SWA1130

VisitID: ET16
Place of interest: Eiffel Tower Group 2

Enter the data into the form and show:

- a screen print of the completed form in **FORM** view
- a screen print of the relevant table showing the new record.

(2)

(f) Use the form you created in Activity 3 to enter and save this choice of visit for Stuart Catterill, StudentID: CAT1118

VisitID: ET15
Place of interest: Eiffel Tower Group 1

Click the Save button and screenprint the message.

(1)

(Total for Activity 4 = 13 marks)

Activity 5 – Printing Reminders (suggested time 1 hour and 30 minutes)

Note: this activity requires you to produce a database report. The activity number, your name, candidate number and centre number should be in the page header for the report. (You need to modify your report in DESIGN view to do this.)

Charles would like to be able to print reminders that show the student details, the total amount they have paid to date and the amount outstanding. The reports will be printed on A4 paper. Charles should be able to cut each sheet into four individual reminders so that they can be handed to individual students.

(a) Create a query that will find students who have not fully paid.

For each student, the query should:

- display the StudentForename and StudentSurname
- calculate the total amount paid to date
- display the outstanding balance.

Screenprint the query in **DESIGN** view. Ensure that the criteria can be seen in full.

(3)

(b) Create a database report based on the query.

Each reminder should include:

- the name of the student, the total amount paid to date and the outstanding balance
- the name of the trip 'Berrymill Secondary School - Paris Trip'
- the words 'Payment Reminder'
- the dates of the trip – 14 March 2012 to 17 March 2012
- today's date
- the message 'The outstanding balance must be received no later than 15 February 2012.'

(7)

(c) Format the report. Ensure that four reminders are printed on each page.

(1)

Screenprint the report in **DESIGN** view.

Print the database report.

Evidence to be submitted for Activity 5.

- A screen print of the query in **DESIGN** view.
- A screen print of the database report in **DESIGN** view.
- The printed database report.

(Total for Activity 5 = 11 marks)

***Activity 6 – Evaluation (suggested time 1 hour)**

You need to evaluate the prototype you have produced.

In a word processed report:

- evaluate your prototype. This must be an evaluation of how well it carries out the tasks identified in the scenario rather than a commentary of what you have done or your own performance in doing it.
- discuss your recommendations for further functionality. In the scenario you were told that your system was to be the first prototype and that it would only carry out certain tasks. In this section of the evaluation describe what else you think the fully functioning version should do.

The Quality of your Written Communication (QWC) will be assessed in this question.

(Total for Activity 6 = 9 marks)

Standard Ways of Working.

All printouts must contain the activity number, your name, candidate number and centre number.

Pages must be securely fastened to the cover sheet and in the correct order.

A minimum font size of 10 should be used for all word processed documents.

(Standard Ways of Working = 2 marks)

TOTAL FOR PAPER = 90 MARKS
