**Assignment Submission Instructions**

|  |
| --- |
| TERM: **Autumn 2019** |
| MODULE**: IY014 Repeat** **Information System Analysis and Design**  |
| ASSIGNMENT TITLE: **Coursework Assessment Element** |
| SUBMISSION DATE: **Last session week 11 (week commencing 2nd December)** |

**Read all instructions very carefully**

1. Your assignment needs to be submitted by yourself on the date given above.
2. You should observe the page count stated on the assignment brief. You will lose marks if this is not followed.
3. Please follow your assignment brief for details of the appropriate layout, font size, spacing and so on.

1. Submissions should be handed in with a cover sheet attached stating:

|  |
| --- |
| **Module code and title** |
| **Tutor name** |
| **Assessment type** |
| **Student Names and T numbers** |
|  |

1. Assignments that are based on practical work (i.e. artwork, computer programs, etc.), must be submitted as **one** hardcopy and **one** CD ROM copy.

You should submit an electronic copy of your work. The submission date and time should be recorded from the submission of the hard copy, either in class or through reception in case of early or late submission.

1. You will need to sign a submission sheet to confirm submission of your work and that all the work is your own. There are serious penalties for submitting work which is not your own, please refer to Academic Misconduct in the Programme Handbook for further information about this.
2. For early/late submission you should fill in a separate cover sheet (available from reception) which also confirms submission and that all the work is your own. You must also collect a receipt as proof that you have submitted your work.
3. The following penalties apply for late submissions:

|  |  |
| --- | --- |
| **Number of Working Days Late** | **Penalty Awarded** |
| 1 | 85% of original mark |
| 2 | 80% of original mark |
| 3 | 75% of original mark |
| More than 3 | Zero mark awarded |

1. If you failed to submit on time due to an Exceptional Extenuating Circumstance (EEC), you should submit an EEC form within three days of the assessment deadline. These are available from reception and may, depending on your circumstances, affect your final mark.

 -

**Assignment Task Description:**

**Sunderland Cruise Ships organises** cruises in Europe, North America and the Far East. During the cruise, the ship docks at various ports. Although passengers are free to explore the ports themselves, the company offers excursions of interest in the port. Once the passengers have booked their cruise, they are in a position to book the excursions related to the cruise. The passengers can either book the excursions online, by phone or in person once the cruise has started. There are a restricted number of places on the excursions and so when they are full up the person goes on a waiting list to see if anyone pulls out of the excursion. Once a person has booked tickets for the excursion their details are stored and they are provided with further information on what will be involved. There will be recommendations on clothing and footwear to bring and whether food is provided. If a person does cancel a booking within 2 weeks of the excursion they will receive a refund minus a £20 admin charge per seat booked.



**Task 1 Normalise Data for the Cruise Ship Excursion Booking System (30 marks)**

**Sunderland Cruise Ships** have heard that they should use a relational data for maintaining the customer details and excursions that are booked. Explain to them what a relational database is. This description should include the advantages and disadvantages of this approach with regards to the system being developed and the data being stored.

Take the preliminary un-normalised table for the Excursion Booking System below and put it into third normal form.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Order ID** | **Passenger** **ID** | **Passenger Name** | **Cabin** | **Cruise No** | **Cruise Name** | **Port**  | **Excursion No.**  | **Excursion** | **Qty** | **Price per excursion** | **Excursion leader ID** | **Excursion Leader**  | **Total Cost** |
| O23 | P001 | Weber | 2345 | 1012 | Baltic Highlights | Copenhagen | C001 | Little Mermaid | 5 | 200 | E0001 | Wermter | 1900 |
|  |  |  |  |  |  | Oslo | O002 | Museums | 2 | 150 | E0002 | Smith |  |
|  |  |  |  |  |  | St Petersburg | P002 | Palaces | 1 | 300 | E1008 | Jones |  |
|  |  |  |  |  |  | St Petersburg | P003 | Ballet | 3 | 100 | E1008 | Jones |  |
| O32 | P005 | Elshaw | 3777 | 2121 | Fjords | Bergen | B001 | Biking | 4 | 50 | E0070 | Malone | 1450 |
|  |  |  |  |  |  | Bergen | B111 | Hiking | 10 | 75 | E0070 | Malone |  |
|  |  |  |  |  |  | Holden | H002 | Puffins | 5 | 100 | E0101 | Ham |  |
| O01 | P003 | Brown | 8124 | 1012 | Baltic Highlights | Oslo | O002 | Museums | 3 | 150 | E0002 | Smith | 1050 |
|  |  |  |  |  |  | St Petersburg | P002 | Palaces | 2 | 300 | E1008 | Jones |  |

Describe what normalisation is and what are its advantages and disadvantages.

You should take the data in the un-normalised form and go through the steps to put the data in third normal form. Ensure you show the tables and keys produced in first, second and third normal form.

**Marking Scheme Task 1**

|  |  |
| --- | --- |
| Achieve 40%  | Achieve Over 70% |
| * A basic description of relational databases with limited consideration of advantages and disadvantages.
* Evidence of partially correct normalisation applied to the identified database relations for achieving each normal form. The relations produced for each normal form should show their attributes and key fields and be populated with some data to illustrate the normal form the database is in.
* For each activity a brief explanation of design decisions should be provided.
* A simple relational model showing the relations (tables) corresponding to the identified entities, their attributes and key fields clearly indicated.
 | * Excellent description of the relational databases including strengths and limitations related to the system and the data stored.
* Evidence of normalisation correctly applied to the identified database relations for achieve each normal form. The relations produced for each normal form should show their attributes and key fields and be populated with data to correctly illustrate the normal form the database is in.
* For each normalisation step an explanation and justification of how and why it was applied.
* A detailed relational model showing the relations (tables) corresponding to the identified entities, their attributes, key fields and any foreign key fields clearly indicated.
 |

**Task 2 Class Diagram (20 Marks)**

Create a class diagram based on the description of the system given above. Show the classes, attributes, methods and the correct relationships that exist between the classes.

**Marking Scheme Task 2**

|  |  |
| --- | --- |
| Achieve 40%  | Achieve Over 70% |
| * A simple class diagram showing some of the required classes, methods and attributes. The diagram shows some appropriate relationships between entities, but it lacks aggregation and composition.
* Some correct use of class notation.
* Limited use of an appropriate CASE tool.
 | * A detailed class diagram showing all the classes correctly identified from the problem description, their attributes, and methods. The diagram shows the correct relationships between all the classes including aggregation, composition and inheritance.
* Uses correctly the notation related to class diagrams.
* Excellent use made of an appropriate CASE tool.
 |

**Task 3 Create the Database (20 marks)**

Using SQL create the tables you design in third normal forms. This will require you to identify the appropriate data types and their sizes, the primary and foreign keys. You should input an appropriate amount of data into these tables. You should show the command in the SQL environment and the outcomes from creating the tables and storing the data.

**Marking Criteria – Task**

|  |  |
| --- | --- |
| Achieve 40%  | Achieve Over 70% |
| * Some evidence of how to create appropriate tables using SQL.
* Some evidence of how to populate tables using SQL.
* Limited understanding of the data types and their sizes.
 | * Clear evidence of how to create appropriate tables using SQL. Including the use of primary and foreign keys, NULL and Non-NULL values and constraints.
* Clear evidence of how to populate tables using SQL.
* Clear understanding of the use of data types and sizes.
 |

**Task 4 Use SQL Command to Adapt Data and Show Information (30 marks)**

In this task you will create the appropriate commands to change the data in the tables and to display information related to the storage of information. You should show the actual commands run in the SQL environment and the output you create.

1. Update the cost for excursions where the port is ‘Copenhagen’ so they are increased by a percentage equivalent to the 4th and 5th characters in your T-Number. For example, if your T-Number is T003456 increase the excursion prices by 34%.
2. Create a list that shows the orderID, passenger number, and passenger name and passenger cabin. The headings for the columns should be “Order ID”, “Passenger ID”, “Name of Passenger” and “Cabin”.
3. Display the passenger details for all passengers going on an excursion in St Petersburg or Oslo.
4. Display the order number, passenger name and who has bought more than 4 tickets on the C001 excursion.
5. If the **last character of your T-Number is greater than 6**, create the SQL to count the number of passengers by excursion leader. Include the excursion leader’s name.

If the **last character of your T-number is less than 6**, produce the SQL to get how many of each excursion are booked. Include the excursion name.

If the **last character in the T-Number is equal to 6**, produce the SQL to determine how many excursions are booked by cruise name.

1. Display the passenger details, excursion details for passengers whose name starts with the first two letters of your first name but do not include the last 2 letters of your last name.
2. Display all excursions where the price in pounds is between the 4th and 5th characters of your T-number and the 6th, 7th and 8th characters of your T-number. For example, if your T-number is T0086299, you should return all excursions whose price is between £86 and £299.
3. Get all the passengers who have booked one or more of the excursions that Weber is on.

**Marking Criteria – Task 4 (30 Marks)**

|  |  |
| --- | --- |
| Achieve 40% | Achieve Over 70% |
| * Some understanding of the updating of tables.
* Completion of some of the simpler SQL test activities.
* An attempt at using aggregate operation, wildcards, embedded selects and joins.
* A report section that show the correct outcomes for some of the SQL statements for activities.
* Little explanation of the SQL statements created and the outcomes from the test activities.
 | * Clear understanding of how to update the SQL tables.
* Evidence of clear understanding of the use of SQL to extract information from a database.
* Successful use of SQL approaches such as aggregates, wildcards, embedded selects and joins.
* Complete all of the SQL test activities.
* A report section with the correct outcomes from all the SQL statements for the test activities.
* Clear explanation of the SQL statements creates and the outcomes for the test activities.
 |

**Marks Breakdown**

**Task 1 Normalisation**

|  |  |
| --- | --- |
| Description of relational database | 10 marks |
| First Normal Form | 5 marks |
| Second Normal Form | 5 marks |
| Third Normal Form | 5 marks |
| Primary and Foreign Keys | 5 marks |

**Task 2 Class Diagram**

|  |  |
| --- | --- |
| Classes | 4 marks |
| Attributes/Methods | 4 marks |
| Relationships | 4 marks |
| Correct Notation  | 4 marks |
| Use of CASE tool | 4 marks |

**Task 3 SQL Develop Database**

|  |  |
| --- | --- |
| Create tables | 4 marks |
| Appropriate datatypes and attributes | 4 marks |
| Constraints | 5 marks |
| Add data | 4 marks |
| Presentation of work | 3 marks |

**Task 4 SQL Data Manipulation**

|  |  |
| --- | --- |
| i) Update the cost for excursions where the port is ‘Copenhagen’ so they are increased by a percentage equivalent to the 4th and 5th characters in your T-Number. For example, if your T-Number is T003456 increase the excursion prices by 34%.  | 1. marks
 |
| ii) Create a list that shows the orderID, passenger number, passenger name and passenger cabin. The headings for the columns should be “Order ID”, “Passenger ID”, “Name of Passenger” and “Cabin”. | 3 marks  |
| iii) Display the order number, passenger name and who has bought more than 4 tickets on the C001 excursion. | 3 marks |
| iv) Display the order number, passenger name and who are buy more than 4 tickets on the C001 excursion. | 3 marks |
| v) If the last character of your T-Number is greater than 6, create the SQL to count the number of passengers by excursion leader. Include the excursion leader’s name. If the last character of your T-number is less than 6, produce the SQL to get how many of each excursion are booked. Include the excursion name. If the last character in the T-Number is equal to 6, produce the SQL to determine how many excursions are booked by cruise name. | 3 marks |
| vi) Display the passenger details, excursion details for passengers whose name starts with the first two letters of your first name but do not include the last 2 letters of your last name.  | 4 marks |
| vii) Display all excursions where the price in pounds is between the 4th and 5th characters of your T-number and the 6th, 7th and 8th characters of your T-number. For example, if your T-number is T0086299, you should return all excursions whose price is between £86 and £299.  | 4 marks |
| viii) Get all the passengers who have booked one or more of the excursions that Weber is on. | 4 marks |
| Presentation of Results and commands | 3 marks |