**ISYS104**

**Assignment Specification**

|  |  |
| --- | --- |
| **Module** | **Assignment 2** |
| **Level** | **Access Databases (Entry-Intermediate Level)** |
| **Due For Completion** | **iLearn Submission, by 28th May, 2017 11.55PM** |
| **Software Used** | Microsoft Access 2010 or later |

### Module Assignment is worth 10 marks

* Technical part: 7 marks (submit on iLearn by 28th May, 2017 11.55PM)
* Discussion part: 3 marks (in workshops - week 12)

In this assignment, you will:

* Create table
* Create relationships (links) between the tables.
* Enter records into the tables.
* Do queries, advanced queries on the database
* Create a form.
* Create a report.

**Task 1: Creating a TABLE (1 mark)**

### (Create TABLE, FIELDS, ENTERING RECORDS, CREATING RELATIONSHIPS)

We have a partially created a database in Access called **ORDERDB**. You need to complete the database by creating CUSTOMER table and fill the table with the records provided.

1. Open the database file called **ORDERDB** (available on iLearn-in Assignment2 Folder).
2. Create a table called **‘CUSTOMER’** with the following data

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Description** | **General Properties** |
| CustomerID | Short Text | Customer ID Code | Size: 4  Format: > |
| CustomerName | Short Text | Name of Customer | Size: 30  Format: > |
| CustomerSuburb | Short Text | Suburb of Customer | Size: 20  Format: > |
| CustomerPostCode | Number | Postcode of the suburb | Size: Integer  Default Value: (leave it empty-delete that 0) |

1. Now you will have to enter records into **CUSTOMER** table and identify the **PRIMARY KEY**

|  |  |  |  |
| --- | --- | --- | --- |
| **CustomerID** | **CustomerName** | **CustomerSuburb** | **CustomerPostCode** |
| CUS1 | HARRY POTTER | EPPING | 2121 |
| CUS2 | RONALD WEASLEY | BURWOOD | 2134 |
| CUS3 | HERMIONE GRANGER | EPPING | 2121 |
| CUS4 | LILY POTTER | EASTWOOD | 2122 |

### Task 2: RELATIONSHIPS (1 mark)

1. Creating Relationship**. [Hint: Go to** **Database Tools –> Relationships]**
2. For each link, make sure you **ENFORCE REFERENTIAL INTEGRITY** in the relationship’s properties section.

### Task 3: QUERIES (1.5 marks – deduct 0.25 marks for each wrong query)

**Note**: If results contain duplicate data, you need to display them only once.

You are now going to **CREATE 🡪 Query Design**:

1. Show customer details (ID, Name and PostCode) who come from a suburb that begins with ‘E’. Suburb must not be displayed in the output. Sort the results by Customer name, in descending order. Save the query as **SUBURB**.
2. Show products’ names and price that costs between $20(exclusive) and $60(inclusive). Sort the results by price, in ascending order. Save the query as **PRODUCTPRICE**.
3. Show values from OrderLine table if the Product ID is ‘PRO1’ or ‘PRO3’ and the quantity ordered is over 8. Save the query as **OVER8**.
4. Show order details (ID, Date, OrderShipped?) for orders that have NOT been shipped. Save the query as **NOTSHIPPED**.
5. Show details of orders (ID, shipment status) along with respective Customer name. Save the query as **CUSNAME.**
6. Show names of products and order date for ‘ORD2’. Save the query as **ORDER2**

### Task 4: Advanced Queries (1.5 mark – 0.75 marks each)

Create the following queries:

For both queries, use **EXPRESSIONS BUILDER** and rename any **NEW** fields with the appropriate field names.

1. Show Order ID, order date, total cost for each order along with CustomerName. Total cost will be OrderQuantity \* ProductPrice, for each product. Sort the results by total cost in descending order. Save the query as **ADVANCEDQ1.**
2. Write an UPDATE Query and increase the prices of products by 5%. Save the query as **UPDATEBY5.**

### Task 5: Form (1 mark)

For this task you will need to

* Create a form for the table CUSTOMER with all fields except Postcode. Choose ‘Tabular’ layout.
* You can use any colours and formatting you like, as well as any design elements
* Include a logo in the form.
* Lock the ‘CustomerID’ field.

### Task 6: Creating a Report (1 marks)

For this task you will need to create a report that includes

* OrderID, OrderDate, ProductID, ProductPrice and Quantity
* View must be based on Orderline table
* Group data by OrderID
* Sort data by OrderDate-descending order
* Choose Stepped layout, Portrait orientation
* You must use the same colours and formatting as the form.