

COMP 1000 Lab 11

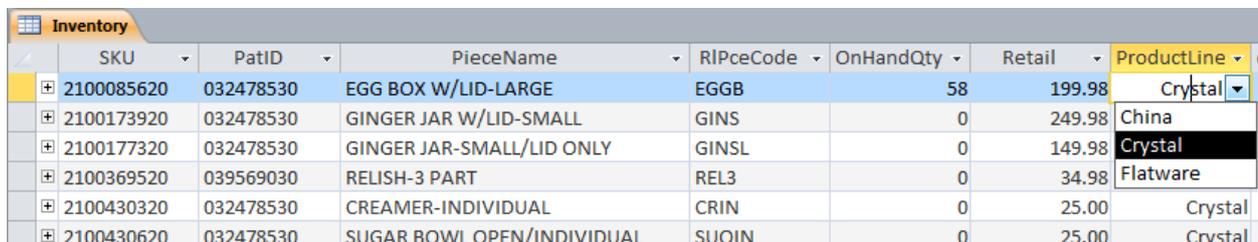
Question 1 (5 marks)

As the database manager at Replacements Ltd, the world's largest selection of old and new dinnerware, your task is to add validation rules, create a form that data entry associates can use to add new items to the inventory and create a new table to classify merchandise as China, Crystal or Flatware. You will also use queries and reports to analyse existing data. Open the database *a06c1replace* and save the database as *a06c1replace_lastName_firstName*

- a. Using Design View, create a new table *ProductLines* with the following fields (1) *ProductLineID* (Data Type – Autonumber) (2) *ProductLineDescription* (Data Type –text, field size=10). Set *ProductLineID* as the Primary Key. Add three records: China, Flatware, Crystal. Close the table.

Adding a lookup field

- b. Create a lookup Field in the *Inventory Table* for the field *ProductLineID* as shown below.



SKU	PatID	PieceName	RIpceCode	OnHandQty	Retail	ProductLine
2100085620	032478530	EGG BOX W/LID-LARGE	EGGB	58	199.98	Crystal
2100173920	032478530	GINGER JAR W/LID-SMALL	GINs	0	249.98	China
2100177320	032478530	GINGER JAR-SMALL/LID ONLY	GINsL	0	149.98	Crystal
2100369520	039569030	RELISH-3 PART	REL3	0	34.98	Flatware
2100430320	032478530	CREAMER-INDIVIDUAL	CRIN	0	25.00	Crystal
2100430620	032478530	SUGAR BOWL OPEN/INDIVIDUAL	SUOIN	0	25.00	Crystal

Use the new table you created above as the source for the values in the lookup field. (**Hint:** Change the Data Type to Lookup Wizard and use the table above for values in the lookup field)

Establishing Validation Rules

- c. Establish Validation Rules on the *Inventory* table as follows:
Set the *SKU*, *OnHandQty*, and *ProductLineID* fields to **Required**.
Establish a validation rule for the *OnHandQty* that requires the value to be atleast zero. The validation text should be: **The value of this field must be greater than 0.**
- d. Open the *Employees* table and add an input mask for the field *Phone*.

Create a validation form

- e. Create a form based on the *Inventory* table with the following rules:
- Delete the subform generated for the *Order* Item table, when you create a form based on the *Inventory* table.
 - Adjust the width of the text box controls until they are 3” wide.
 - Apply the **CurrencyFormat** to the *Retail* field
 - Add a sunken special effect to the text box controls
 - Delete the form icon image in the title. Add necessary spaces to the field labels.
 - Save the form as **Inventory**. Demonstrate that the validation rules work.
 - Create a copy of the new Inventory form and name it **Lookup Inventory Form**. Change the properties of the new form so that users cannot add, edit, or delete records. (**Hint:** Change the **Allow Edits** and other necessary properties in the Property Sheet to **No**)
 - Demonstrate that you cannot add, delete or edit records in the lookup form

Adding a data macro

- f. Attach a data macro to the Before Change event of the Inventory table. Add actions that will check for inventory items with a retail value greater than \$5000.00. Display an appropriate message to alert the user, *Verify the retail value is correct*. Test the data macro.

Question 2 (5 marks)

Northwind Traders provides specialty foods to businesses around the world. Its nine employees are looking for ways to be more efficient when processing orders. You have been asked to filter the orders so each employee can see only his or her orders. You will also filter a report using an SQL statement.

- a. Open the database *a10m1traders* and save as *a10m1traders_lastnameFirstname*.
b. Open the *Orders* form. Advance through the form using the Navigation Bar. Notice the *EmployeeID* value is different for each order.

Order Processing Form

Employee filters. Click 'x' to clear all filters.

1 2 3 4 5 6 X

OrderID: 10248
CustomerID: WILMK
EmployeeID: 5
OrderDate: 4/28/2007
RequiredDate: 6/10/2006
ShippedDate: 5/26/2007

OrderDetails

Product	Quantity	UnitPrice	Ext Price
Queso Cobrales	12	\$21.00	\$252.00
Singaporean Hokkien Fried Mee	10	\$14.00	\$140.00
Mozzarella di Giovanni	5	\$34.80	\$174.00

Record: 1 of 3 | No Filter | Search

Record: 1 of 830 | Unfiltered | Search

- c. Add a button to the form with caption **1** as shown in the picture above. Click **Cancel** when the Command Button Wizard appears. Reduce the size of the button to match the button shown.
- d. Link this button to a **On Click Event**, such that when the button is clicked, it applies a filter to display only orders belonging to **EmployeeID = 1**. [**Hint:** Using the property sheet, associate an On Click Event to the button. Click **Build**, choose Macro Builder. Type **ApplyFilter** in the **Add New Action box**. Type **[EmployeeID] = 1** in the **Where Condition box**. Save and close the macro.]. (Please note, hints will not be provided in the exam.). Test this new button. Clicking on this button should now display 123 orders instead of 830 total orders shown previously. Verify that the *EmployeeID* is 1 for each order.

- e. Add buttons with captions 2 to 6 using the same step as above to filter orders based on different *EmployeeIDs*. Create a seventh button (caption X and background color 'red') to remove the filters so all the orders will display in the form. [Hint: Choose **RemoveFilterSort** in the **Add New Action Box**] Test each new button when finished.
- f. Open the customers report. Edit this report to remove to delete the *Postal*, *Phone*, and *Fax* fields and their labels, adjust the width of the *Country* field, and customize the report to print customers from a particular country. The report should prompt for the user to enter the name of the country. [Hint: Open the Property Sheet, select the **Report**, and then edit the **Record Source Property**. Type in *SELECT * from CUSTOMERS WHERE country = [Enter a country]* statement in the record source in place of the *CUSTOMERS* table. Test your report for different countries.

Question 3 (5 marks)

You need to prepare a reports showing employee raises and bonuses by city for the Human Resources department of the Comfort Insurance Agency as shown in the figure below.

Location	LastName	FirstName	HireDate	Salary	2012Increase	2012Raise	Year1
Los	Abrams	Wendy	5/24/2012	\$47,500.00	3.00%	1425	
	Anderson	Vicki	9/21/2008	\$47,900.00	4.00%	1916	
	Bchette	Susan	9/10/2012	\$61,500.00	4.00%	2460	
	Block	Leonard	12/23/2010	\$24,200.00	3.00%	786	
	Brown	Patricia	6/12/2011	\$20,100.00	5.00%	1005	
	Brumbaugh	Paige	11/25/2009	\$49,300.00	3.00%	1479	
	Daniels	Phil	1/5/2011	\$42,600.00	3.00%	1278	
	Davis	Martha	6/14/2010	\$51,900.00	4.00%	2076	
	Drubin	Lolly	9/11/2009	\$37,000.00	3.00%	1110	
	Fantis	Laurie	1/11/2011	\$28,000.00	3.00%	840	
	Fleming	Karen	11/15/2009	\$41,100.00	3.00%	1233	
			12/31/2008	\$38,400.00	3.00%		

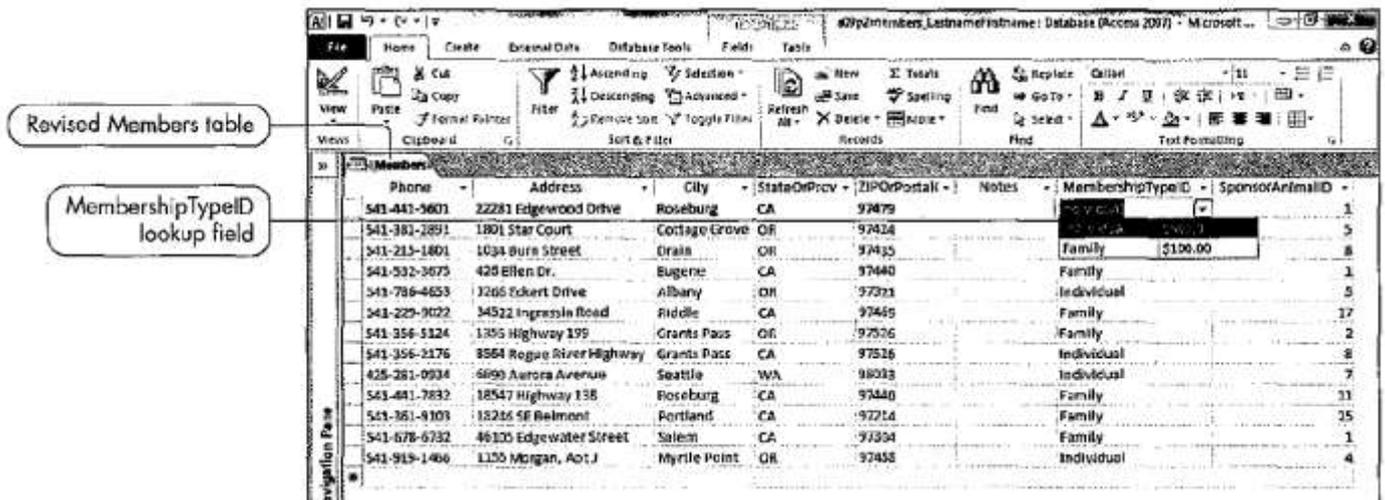
- a. Open the database *a04p2insurance* and save as *a04p2insurance_lastnameFirstname*.
- b. Create a **Form** based on the *Locations* table and apply a **Solstice Theme** for this form object. Reduce the size of the text box *LocationID* to approximately 50% of its original size. Save the form as *Locations*.
- c. Next, create a **Report** based on the *Locations* table. Reduce the size of the *Locations*, *Zipcode*, and the *OfficePhone* to 50% of its original size. Adjust the other columns if necessary until there are no controls on the right side of the vertical dashed line. Save the report as *Locations*.
- d. Next using the **Report Wizard**, create a report based on the *Employee Query*. The following criteria must be addressed:
- Apply **Theme Module** theme to this report object.
 - Employees must be grouped by *Location*
 - Use *LastName* as the first sort order and *FirstName* as the second sort order.
 - The report must calculate the following fields:

- **Sum** for *Salary*,
- **Average** for *2012Increase*,
- **Average** for *YearsWorked*. (Format: Fixed, Decimal places 2)
- Report must use a **Stepped** layout.
- Orientation is Landscape.
- Title of the report should be **Employee Compensation**
- Change the **YearsWorked** label to **Years**.

e. Display the report in Print Preview. Save and close the Employee Compensation Report.

Question 4 (5 marks)

The metropolitan zoo invites its patrons to become members of the zoo. For a donation of \$50 per year for an individual, or \$100 for a family membership, members are entitled to special discounts and member only promotions. Your task is to review the Members database to review the tables for normalisation errors, to check the performance, to create a lookup field and then save the database



- Open the database *a10m1members* and save as *a10m1traders_lastnameFirstname*.
- Open the members table, examine the data and determine if the data meets the three normalization rules: 1NF, 2NF, 3NF
- You decide to normalize this table manually.
- Using the **Query Design**, design a select query to select the fields *MembershipTypeID*, *MembershipTypeName*, *MembershipDonationFields* from the *Members* table. Run the query. Take note of the repeating rows.
- Use the above query to create a *Make Table Query*, so that the two records are copied to a new table called *MemberTypes*. Save the query as *Create Member Types*. (You have just created a second table to help normalize the Members table). Make the field *MembershipID* as the primary key for the new table.
- Edit the *Members* table, to change the data type of the field *MembershipTypeID* to be a lookup field using the new table *MemberTypes* [Hint: Use the Lookup Wizard]. Verify that the lookup field is working.
- Next delete the redundant fields *MembershipTypeName* and *MembershipDonation* fields from the *Members* table.
- Next, create a form for each of the three tables in the database using the Form tool. Delete any subforms that appear automatically.
- Create a report based on the Members table using the **Report** tool. Remove a few fields so that the remaining fields fit into one page.

- j. Use these four new objects to create a **Navigation Form** using the Horizontal tabs template (see figure above). Edit the fourth tab and change the caption to **Members Report**. Test the new Navigation form.
- k. Click the Database Tools tab, then click on Analyse Performance. Click **All Object Types** tab, click **Select All**, and then click **OK**. The first idea in the Results window suggests you save your application as an MDE file.