

# FINAL EXAM

## INSTRUCTIONS:

1. This exam is 20% of your final grade. It is intended to test major topics discussed during the previous 5 weeks. Each question is independent of the others, so if you are unclear on what to do, you may skip it and continue on.
2. This is an open book, open note exam. You may **not** ask for help from another student.
3. This exam is due by December 16 at 11 PM CST. It should be uploaded to Moodle under the Assignment labeled "Final Exam."

---

### Files needed for the Assignment:

- **Always.accdb**

### Files to Upload to Moodle:

- **Always.accdb**

Chris and Pat Dixon own and manage Always Connected Everyday (ACE), a successful Internet service provider (ISP) in your area. ACE provides Internet access to residential and business customers and offers a variety of access plans, from dial-up and DSL to wireless. Within each type of service-dial-up, DSL, and wireless-ACE offers low-cost plans with either slower access speeds or fewer capabilities and more expensive plans with either higher access speeds or greater service and features.

To keep track of their business, Chris and Pat have developed the Always database. The database has two tables: the **tblAccessPlan** table contains data about the plans they offer commercial and residential customers, and the **tblCustomer** contains data about their customers. The database also contains several queries, forms, and reports.

1. Open the **Always** database provided with your Data Files.
2. Use your first and last names for the first record in the **tblCustomer** table.
3. Chris and Pat have provided the data in this table to show the kind of service information they would like to start tracking in their database.

CustomerAcctNum	Service Description	Service Rate	Service Date
002-00300	Dial-up install	\$35.00	04/12/2012
081-01001	DSL onsite	\$50.00	04/17/2012
081-01001	DSL repair	\$50.00	04/18/2012
081-86410	DSL repair	\$50.00	04/20/2012
081-00100	Wireless business corrective	\$100.00	04/19/2012
081-00100	Wireless business corrective	\$100.00	04/20/2012

Upon studying the data, you can see that it is not normalized. There is repeating data for both Customers and Service Descriptions. You will need to create two new tables for this assignment and define a many-to-many relationship. Your new tables should include the following:

- a. The **tblService** table includes a unique service ID (not specified in the data), a service description, and a service rate.

- b. The **tblServiceCall** table includes, the customer account number, service ID, and service date. **HINT: Customers can have the same Services repeatedly (but always on different dates) – so think carefully about how many fields are necessary for a primary key here.**
- c. For each table, ensure you have the following in place to receive full credit:
- Correct primary key(s).
  - Appropriate and meaningful names for the table and the fields.
  - Correct data types for each field.
  - Database relationships with enforced referential integrity and Cascade Updates turned on.
  - Data from the table entered into your table correctly.
4. Chris and Pat have also discovered that a few of their customers have Billing Addresses in addition to their service address. The Billing information is as follows:

Customer Acct Num	Billing Address
002-00200	PO Box 999, Drayton, MI 49853
081-20228	PO Box 3521, Drayton, MI 49853
002-59547	237 N Ridge Blvd, Blade, MI 49421

- Rather than normalize as we have previously, they would rather create a separate table just for Billing Addresses with a one-to-one relationship. Create a new table called **tblBillingAddress** that includes the data above. Remember that it might be necessary to breakdown some of the fields into multiple columns in your Access table. For your table, ensure you have the following in place to receive full credit:
- Correct primary key.
  - Appropriate and meaningful names for the fields.
  - Correct data types for each field.
  - One-to-one relationship applied with correct integrity options.
  - Data from the spreadsheet entered into your table correctly.
5. Create a custom report based on the qryAccessPlanCustomers query. **Figure 1** shows a sample of the completed report. Refer to the figure as you create the report. Save the report as **rptAccessPlanCustomers**.
- The report will require Grouping on one field.
  - The report will require Sorting on two fields.
  - Setting the Alternate Row Color correctly is optional, but will garner bonus points if successfully completed.
  - Be sure to include a title for all pages.
  - The summary values are Count for LastName and Average for PlanMonthlyCost. The labels shown to explain that on the report are worth extra credit and are not required.
  - Be sure to include the Date and page numbers as shown.

6. Create a query called **qryCustomerAccessPlanCost** that contains the fields CustomerAcctNum, CompanyName, FirstName, LastName from the tblCustomer table and AccessPlan from the tblAccessPlan table. Add a computed field named **AnnualizedMonthlyCost** that multiplies the PlanMonthlyCost by 12 to calculate the total price of the plan for a year. Sort in descending order by PlanAnnualCost, AccessPlan Ascending.
7. Create a query called **qrySeptANDBlade** that displays the FirstName, LastName, NextBillingDate, and Email Address for customers who have Next Billing Dates in the month of September 2010 and live in Drayton. Sort in ascending order by LastName, FirstName.
8. Create a make-table query based on the tblCustomer table, selecting the CustomerAcctNum, CompanyName, FirstName, LastName, City, and AccessPlanID fields from the table. Select those records with a City field value of **Blade** or **Drayton**. Save the table as **tblSelectedCustomer** in the current database. Save the query as **qmakSelectedCustomer**. Run the query and then close it.
9. Create a query with the tblAccessPlan and tblCustomer tables, selecting all records from the tblAccessPlan table and any matching records from the tblCustomer table. Display the AccessPlanID, AccessPlan, and PlanMonthlyCost fields from the tblAccessPlan table, and the CompanyName, FirstName, and LastName fields from the tblCustomer table. Save the query as **qryPlanCustomerOuterJoin**, and then run and close the query.
10. Create the following indexes. Give them appropriate names.
  - a. Email - in ascending order in tblCustomer and allows duplicates.
  - b. Zip – in ascending order in tblCustomer and allows duplicates.
  - c. PlanMonthlyCost - in descending order in tblAccessPlan.

FIGURE 1:

Access Plan Customers

12/10/2012

City	First Name	Last Name	Plan Monthly Cost
Dial-up unlimited			
Brunson	Xinyan	Jiang	\$19.95
Brunson	William	Walker	\$19.95
Drayton	Hava	Jukic	\$19.95
# Customers: 3		Avg Monthly Cost	\$19.95
DSL business			
Drayton	Edward	Petrosyan	\$89.95
Drayton	Brad	Visser	\$89.95
Grand Haven	Michelle	Bousu	\$89.95
Grand Rapids	Kristine	Cook	\$89.95
# Customers: 4		Avg Monthly Cost	\$89.95
DSL pro			
Blade	Jorge	Suarez	\$29.95
Blade	Mayra	Uzarski	\$29.95
Brunson	Rosalinda	Ramirez	\$29.95
# Customers: 3		Avg Monthly Cost	\$29.95
Wireless city basic			
Blade	Amy	Zainea	\$49.95
Grand Haven	Marcia	Crane	\$49.95
# Customers: 2		Avg Monthly Cost	\$49.95
Wireless city business			
Drayton	Francisco	Ayala	\$99.95
Grand Rapids	Valencia	Sanchez	\$99.95
Grand Rapids	David	Styne	\$99.95
# Customers: 3		Avg Monthly Cost	\$99.95
Wireless city economy			
Blade	Jacob	Koppelman	\$39.95
Drayton	Khue	Do	\$39.95
Drayton	Joy	Eakins	\$39.95
# Customers: 3		Avg Monthly Cost	\$39.95
Total Customers: 18		Avg Monthly Cost	\$57.17