**BITS Corporation Module 4**

1. Create a view named TopLevelCust view. It consists of the number, name, address, balance, and credit limit of all clients with credit limits that are greater than or equal to $10,000. Display the data in the view.
2. Create a view named ItemOrder view. It consists of the TaskID, description, price, order number, order date, and quoted price for all order lines currently on file. Display the data in the view.
3. Create the following indexes. If it is necessary to name the index in your DBMS, use the indicated name.
   1. Create an index named ItemIndex1 on the TaskID field in the OrderLine table.
   2. Create an index named ItemIndex2 on the Description field in the Tasks table.
   3. Create an index named ItemIndex3 on the Description and Category fields in the Tasks table.
   4. Create an index named ItemIndex4 on the Description and Category fields in the Tasks table and list Category in descending order.
4. Drop the ItemIndex3 index from the Tasks table.
5. Assume the Client table has been created, but there are no integrity constraints. Create the necessary integrity constraint to ensure that the only allowable values for the CreditLimit field are 2500, 5000, 7500 or 10000. Ensure that the ClientNum field is the primary key in the Client table, and foreign key in the WorkOrders table.
6. Because BITS is about to obtain client number 1000, increase the length of the ClientNum field in the Clienttable to four characters. Insert yourself as client number 1001 with sample data. Display all the data in the Client table.
7. Add a field named TimeAllocation to the Tasks table. The allocation is a number representing the number ofminutes that have been initially allocated for the task. Set all Allocation values to 60, as the company has a one hour minimum charge. Display all the data in the Tasks table.
8. Delete the Time Allocation field from the Tasks table. Display all the data in the Tasks table.
9. What command would you use to delete the Tasks table from the BITS database? (Do not delete the Tasks table.)
10. If you are using Access 2016, do the following.
    1. Add a field to the Client table named AvailableCredit. Set the field to currency.
    2. Create a data macro associated with the After Update event for the Client table. When a user updates the Balance field, the macro should subtract the new balance from the credit limit and place that value in the AvailableCredit field.
    3. Create a data macro associated with the After Delete event for the OrderLine table. When a user deletes a record because it is complete or cancelled, the macro should look up the order number in the WorkOrders table and delete it there as well. Use the For Each Record in the WorkOrders table. Hint: You can use the OLD keyword in the Where Condition: [Old].[OrderNum] = [WorkOrders].[OrderNum]. Test the data macro by deleting a record from the OrderLine table and ensuring that the record also is deleted from the WorkOrders table.
11. Using Access 2016, an employee at BITS tried to delete TaskID PI54 from the Tasks table and received the following error message: “The record cannot be deleted or changed because table ‘OrderLine’ includes related records.” Why did the employee receive this error message? What change is needed in the database to allow the deletion of records from the Tasks table?
12. BITS has decided to include I.T. training in its service line and has assigned the item to the category OTH. What change is needed in the database to add items in category OTH to the Tasks table? Would you add any integrity constraints to the fields in the Tasks table? Why or why not?