SQL Assignment

Attempt all the question given below. Put all your queries in a .SQL file and send it to Trainer.

Following Points should be ensured while creating SQL Scripts.

1) Name of the SQL file should be meaningful.
2) Each SQL Statement should be closed with semicolon.
3) Whenever required proper use of BEGIN & END should be used.
4) All the SQL keywords / Reserved Words should be written in Upper Format.
5) For readability, proper indenting of the statements should be done in all the SQL Scripts created.
6) Procedure should be created with proper meaningful names.

Day 1

Q.1) Write a Small Description on System Databases. Create .SQL File for it.

Q.2) Create Database Training

Q.3) Create Schema "Learning", All the further assignment's exercised need to be completed in Database = Training & Schema = "Learning"

Q.4) Create Following Table and create SQL Script for the same.

Table Name : LearnDataType			
Column Name	**Data Type**	**Data Length**	**Constraint**
IDColumn	INT		Not Null
Column1	Char	10	Allow Null
Column2	Nvarchar	10	Allow Null
Column3	BigInt		Not Null
Column4	DateTime		Not Null
Column5	Date		Allow Null
Column6	timestamp		NOT NULL
Column7	Nvarchar	max	Not null

Q.5) Create an INSERT script to Insert data in above table.

Q.6) Create Script to understand **NOT NULL** constraint. Script should contain following (Filename should be Notnull.sql)

1) Create TABLE
2) Write Insert Statements showing success & error.

Q.7) Create Script to understand Check constraint. Script should contain following (Filename should be Check.sql)

1) Create TABLE
2) Write Insert Statements showing success & error.

Q.8) Create Script to understand Default constraint. Script should contain following (Filename should be Default.sql)

1) Create TABLE
2) Write Insert Statements showing success & error.

Q.8) Create Script to understand Unique constraint. Script should contain following (Filename should be Unique.sql)

1) Create TABLE
2) Write Insert Statements showing success & error.

Q.9) Create Script to understand Primary Key constraint. Script should contain following (Filename should be Primarykey.sql)

1) Create TABLE
2) Write Insert Statements showing success & error.

Day 2

Q.1) Script to create following tables.

Table Name: Agent

Column Name	**Data Type & Data Length**
AgentID	**Int , Primary key**
Agent_Code	Char(6) Not Null, Unique Constraint
Agent_Name	Char(40) Not Null
Working_Area	Char(35)
Commission	Decimal (10,2)
Phone_No	Char(15)
Country	Varchar(25)

Create script to Insert following data to the tables.

AGENT_CODE	AGENT_NAME	WORKING_AREA	COMMISSION	PHONE_NO	COUNTRY
A001	Subbarao	Bangalore	0.14	077-12346674	
A002	Mukesh	Mumbai	0.11	029-12358964	
A003	Alex	London	0.13	075-12458969	
A004	Ivan	Torento	0.15	008-22544166	

A005	Anderson	Brisban	0.13	045-21447739	
A006	McDen	London	0.15	078-22255588	
A007	Ramasundar	Bangalore	0.15	077-25814763	
A008	Alford	New York	0.12	044-25874365	
A009	Benjamin	Hampshair	0.11	008-22536178	
A010	Santakumar	Chennai	0.14	007-22388644	
A011	Ravi Kumar	Bangalore	0.15	077-45625874	
A012	Lucida	San Jose	0.12	044-52981425	

Table Name: Customer

Column Name	**Data Type & Data Length**
CustID	**Int, Primary Key Not Null**
Cust_Code	Varchar(6) Not Null
Cust_Name	Varchar(40)
Cust_City	Char(35)
Working_Area	Varchar(35)
Cust_Country	Varchar(20)
Grade	int
Opening_Amt	Decimal(12,2)
Receive_Amt	Decimal(12,2)
Payment_Amt	Decimal(12,2)
Outstanding_Amt	Decimal(12,2)
Phone_No	Varchar(17)
AgentID	Int, Foreign key to Agent Table (AgentID

Create script to Insert following data to the tables.

CUST_CODE	**CUST_NAME**	**CUST_CITY**	**WORKING_AREA**	**CUST_COUNTRY**	**GRADE**	**OPENING_AMT**	**RECEIVE_AMT**	**PAYMENT_AMT**	**OUTSTANDING_AMT**	**PHONE_NO**	**AgentID**
C00001	Micheal	New York	New York	USA	2	3000	5000	2000	6000	CCCCCCC	8
C00002	Bolt	New York	New York	USA	3	5000	7000	9000	3000	DDNRDRH	8
C00003	Martin	Torento	Torento	Canada	2	8000	7000	7000	8000	MJYURFD	10
C00004	Winston	Brisban	Brisban	Australia	1	5000	8000	7000	6000	AAAAAAA	11
C00005	Sasikant	Mumbai	Mumbai	India	1	7000	11000	7000	11000	147-25896312	2
C00006	Shilton	Torento	Torento	Canada	1	10000	7000	6000	11000	DDDDDDD	10
C00007	Ramanathan	Chennai	Chennai	India	1	7000	11000	9000	9000	GHRDWSD	10

C00008	Karolina	Torento	Torento	Canada	1	7000	7000	9000	5000	HJKORED	10
C00009	Ramesh	Mumbai	Mumbai	India	3	8000	7000	3000	12000	Phone No	2
C00010	Charles	Hampshair	Hampshair	UK	3	6000	4000	5000	5000	MMMMMMM	9
C00011	Sundariya	Chennai	Chennai	India	3	7000	11000	7000	11000	PPHGRTS	10
C00012	Steven	San Jose	San Jose	USA	1	5000	7000	9000	3000	KRFYGJK	12
C00013	Holmes	London	London	UK	2	6000	5000	7000	4000	BBBBBBB	3
C00014	Rangarappa	Bangalore	Bangalore	India	2	8000	11000	7000	12000	AAAATGF	1
C00015	Stuart	London	London	UK	1	6000	8000	3000	11000	GFSGERS	3
C00016	Venkatpati	Bangalore	Bangalore	India	2	8000	11000	7000	12000	JRTVFDD	7
C00017	Srinivas	Bangalore	Bangalore	India	2	8000	4000	3000	9000	AAAAAAB	7
C00018	Fleming	Brisban	Brisban	Australia	2	7000	7000	9000	5000	NHBGVFC	11
C00019	Yearannaidu	Chennai	Chennai	India	1	8000	7000	7000	8000	ZZZZBFV	10
C00020	Albert	New York	New York	USA	3	5000	7000	6000	6000	BBBBSBB	8
C00021	Jacks	Brisban	Brisban	Australia	1	7000	7000	7000	7000	WERTGDF	11
C00022	Avinash	Mumbai	Mumbai	India	2	7000	11000	9000	9000	113-12345678	2
C00023	Karl	London	London	UK	0	4000	6000	7000	3000	AAAABAA	6
C00024	Cook	London	London	UK	2	4000	9000	7000	6000	FSDDSDF	6
C00025	Ravindran	Bangalore	Bangalore	India	2	5000	7000	4000	8000	AVAVAVA	11

Q.2) Create Script for following Select Statements.

1) List all the customers leaving in Bangalore.
2) List all the customers leaving in Canada & UK
3) List all the customers leaving not in India
4) List all the customer whose customer name starts with M
5) List all the customer whose opening amount is greater than 7000
6) List all the customer whose opening amount is between 4000 & 6000
7) List all the customer whose Opening Amount + Receive Amount – Payment Amount is greater Than 3000
8) List all the data for agent whose customer opening amount is greater than 5000
9) List all the agent with total customer with each agent. Total customer columns should show 0 if no customer is present for the agent.

Agent Name	Total Customer
Subbarao	5

Anderson	10

Q.10) List all the agent & Customer data whose customer are leaves in (Mumbai or Bangalore) and opening amount is greater than 5000

Q.3) Create script for following Update Statements.

1) Update all agents commission = **0.18** whose customers outstanding amount is greater than **8000.**
2) Update phoneNo = **99999** whose customer name starts with **M** and agent name is **Alford**
3) Update all the working area value in customer table same as the working area value of their agents.

Q.4) Create script for following Delete Statements.

1) Delete all customer records whose agent name is **Alex.**
2) **Delete all customer records whose outstanding amount is less than 5000**

Q.5) Create script for following Truncate Statements.

1) Truncate all the records from agent.
2) Truncate all the records from Customer.

Q.6) Again run Insert scripts created earliers.

Day 3 & Day 4

Q.1) Create following Views

1) Create a view to get customerCity & Customer Name from customer table. View results Should be as below

CustomerCity	CustomerName
New York	Micheal, Bolt, Albert

Hint : Make a use of Outerapply & stuff function.

2) Create a view to get combined data from customer & agent table. View should include following columns
    a. AgentName
    b. CustomerName
    c. CustomerCity
    d. CustomerCuntry
    e. OutstandingAmount
    f. AgentCommission

Q.2) Creating Procedures

1) Create Procedure to Insert data in Agent Table. Pass the value as an Input Parameter to Stored Procedure.
2) Create Procedure to Insert Data in Customer Table. Pass the value as an Input Parameter to Store Procedure.
3) Create Procedure to Update data from Agent Table. Pass the value as an Input parameter to stored procedure. Update to be done based on AgentID.
4) Create procedure to Delete data from customer table. Pass the value as an Input Parameter to stored procedure. Delete to be done based on customerID
5) Create single procedure to Insert, Update & Delete data in agent table. Pass all the required values as an input parameter to stored procedure. Update & Delete to be done based on AgentID.

6) Create single procedure to Insert, Update & Delete data in Customer Table. Pass all the required values as an input parameter to stored procedure. Update & Delete to be done based on customerID.

7) Create procedure to select Customer / Agent records as per the Input Parameter.

    Input Parameter: AgentID, CustomerID, AgentName, CustomerName.

    If AgentID is passed then agent data to output as per the agentid
    If AgentName is passed then agent data to output as per the agentname passed.
    If CustomerID is passed then customer data to output as per the customerid
    If CustomerName is passed then customer data to output as per the customerName.

Q.3) Create a scalar value function to calculate commission for each agent.

Input parameter: Customer ID
Output Parameter: Decimal Value
Calculation: Based On Customer Id, Get AgentID, Based on AgentID get Commission value from Agent Table and return it.

To test the function write Select query to get below listed column.

1) CustomerID
2) CustomerName
3) AgentName
4) OutstandingAmount
5) Commission = OutstandingAmount * Value returned by function

Q.4) Create Table Value Function to get list of all agent.

To test the function write select query to get below listed column

1) CustomerID
2) CustomerName
3) OutstandinAmount

4) AgentName
5) AgentCode
6) Commission

Hint: Join table value function with Customer table on Customer.agentID = <TableValueFunction>.AgentID

Q.5) Create NonClustered index on CustomerTable
Key Columns: CustomerCode, CustomerName
Included Columns: OpeningAmount, Outstanding Amount

Q.6) Create NonClustered Index on AgentTable
Key Columns: AgentCode
Included Columns: AgentName, Commission

Q.7) Create Auditing Mechanism through Insert / Update & Delete triggers on agent table.

1) Create AuditAgentTable : Replica of the AgentTable + following two columns
TransactionDate : This should store date on which transaction was done on agent table
TransactionType : This should store Transaction Type like Insert / Update or Delete.