Assignments

Table of Contents

S#	Session	Page No.
1.	Introduction to Data Integrity	3
2.	Introduction to Indexes	5
3.	Types of Indexes	7
4.	Maintaining Indexes	9
5.	Implementing Views	11
6.	Managing Views	13
7.	Introducing Stored Procedures	15
8.	More About Stored Procedures	17
9.	Introduction to Triggers	18

Introduction to Data Integrity

Sr. No. **Assignment Question**

1

Saint Clara Insurance (SCI) services is a leading Insurance company based in New York, USA. SCI Services wanted a faster, more accurate, and less expensive way to handle insurance claims adjusting for its insurance company customers. With an ever increasing customer base, they decided to create a web based application that will be used not only by employees who work on field but will also be used by the administrators in the head office.

SCI handles approximately 650 claims per month, but that can soar to 15000 or more when a hurricane or some other disaster strikes. Officers can use the software on the device type of their choice: Tablet PCs or laptops in the field, or desktop PCs back in their offices. The use of Microsoft SQL Server 2005 as the software's database enables to receive and update all the necessary information regarding a customer or claimer.

With thousands of customers expected every month, data integrity of the data in the database is very important.

1. Create a database called SaintClaraServices to store the details of the company. Create a table CustomerHeader with the following details.

- **CustomerHeader Table**:

Field Name	Data Type	Key Field	Description
ClientID	Int	Primary Key	Stores client id
FirstName	Char		Stores first name of the client
LastName	Char		Stores last name of the client
MiddleName	Char		Stores middle name of the client
Gender	Char		Stores gender of the client
DateOfBirth	DateTime		Stores date of birth of the client
Address	Varchar(max)		Stores address of the client
MaritalStatus	Char		Stores marital status of the client
Age	Int		Stores age of the client
Employment	Char		Stores occupation of the client
CompanyName	Varchar(max)		Stores the company name
CompanyAddress	Varchar(max)		Stores the company address

Table 1.1: CustomerHeader Table

- **CustomerDetails Table**:

Field Name	Data Type	Key Field	Description
ClientID	Int	Primary Key	Stores client id
FatherName	Char		Stores the name of the client's father
MotherName	Char		Stores the name of the client's mother
Amount	Money		Stores the principal amount
Period	Int		Stores period for insurance
Plan	Char		Stores plan for insurance
Premium	Money		Stores premium
NomineeName	Char		Stores nominee name
Date	DateTime		Stores the date on which insurance is made

Table 1.2: CustomerDetails Table

2. Create a table CustomerHeader with the specifications given in Table 1.1. Identify the primary key for this table so that table entries are not duplicated.

3. Write a query to add a foreign key to CustomerDetails table.

4. Write a query to accept only two values in the Gender field, 'M' and 'F' and also make the default value of MaritalStatus field as 'Single'.

5. Only customers over the age of 21 are eligible to open an insurance policy at SCI. Write a query, so that customers under the age of 21 are not allowed to make an entry to the database.

Introduction to Indexes

Sr. No.	Assignment Question

1 **Houston State Library** is one of the renowned libraries in Houston, Texas. The library has a stock of around 10,00000 books of different genres. The library issue books to the students of the college nearby. With the inflow of students coming to the library growing exponentially, Houston State Library has decided to automate the entire process of issuing books to the students. The library has increased the quantity of each book by 10 copies, depending upon the demand made by the students.

1. Create a database named HoustonStateLibrary to store the details of books in the Library. Create a table named BooksMaster to store the details of the books in the library.

   ➢ **BooksMaster**:

Field Name	Data Type	Key Field	Description
BookCode	Varchar(50)	Primary Key	Stores book code of the book
Title	Varchar(MAX)		Stores the book title
ISBN	Varchar(50)		Stores the ISBN of the book
Author	Char(30)		Stores author name of the book
Price	Money		Stores price of the book
Publisher	Char(30)		Stores publisher name of the book
NoOfPages	Numeric(10,0)		Stores number of pages in the book

Table 2.1: BooksMaster Table

2. Create a table named StudentMaster to store the details of the students who issue a book from the library. Follow the specifications in the table shown below:

   ➢ **StudentMaster**:

Field Name	Data Type	Key Field	Description
BookCode	Varchar(50)	Primary Key	Stores book code of the book
MembershipNo	Varchar(10)	Primary Key	Stores the membership number
Name	Char(30)		Stores the name of the student
Age	int		Stores age of the student
Address	Varchar(MAX)		Stores address of the student
DateOfIssue	DateTime		Stores date of issue of the

			book
DateOfReturn	DateTime		Stores date of return of the book
ISBN	Varchar(50)		Stores the ISBN of the book
Title	Varchar(MAX)		Stores the book title

Table 2.2: StudentMaster Table

3. Create a suitable primary key for the table BooksMaster. Ensure that there is a unique book code for every book. Books with similar title and author but with a different book code and a different ISBN number can be entered into the table.

4. Create a foreign key for the table StudentMaster. Use BookCode as the foreign key for the StudentMaster table.

5. Create a clustered index named IX_Title on the Title column in the BooksMaster table.

6. The Houston State Library Management wants to track the number of books issued to a particular student. Create a nonclustered index IX_MemberNo on the table StudentMaster table.

Types of Indexes

Sr. No. **Assignment Question**

1

Pan World Tours and Travels, founded in 1998, is one of the most popular travel agencies in the heart of North America. This travel agency has built a strong reputation as an outstanding travel agency in terms of location and advertising visibility. As the company's reputation has grown, it has successfully entered several American markets. It offers travelers a chance to discover the marvels of western Canada, and American sights as Yellowstone Park, the Grand Canyon, Zion National Park, Bryce Canyon National Park, and Alaska.

Pan World Tours and Travels operates all year round, offering programs that are customized to the various seasons. In addition to its regular tours, Pan World Tours and Travels now offers visitors the facility of booking and staying in luxurious hotels of America.

Hence, to maintain the growing hotel booking business, a database is required for a smoother and easier operation. The database should have the following details:

1. **Hotel**: This unit gives the list of hotels in different cities of United States of America.

2. **Room**: This unit is involved with all the room information, along with their prices for the different types of rooms in the list of hotels provided.

3. **Booking**: This unit is mostly concerned with all the booking related details for the rooms of corresponding hotels.

4. **Tourist**: This lists out all the required information about the tourists who have booked the hotels through Pan World Travel agency.

Create a database named PanWorld with the following tables:

- **Hotel table**:

Field Name	Data Type	Key Field	Description
HotelNo	Int	Primary Key	Stores the hotel identification number
HotelName	Char		Stores the hotel name
City	Char		Stores the city name
Address	Varchar(MAX)		Stores the address

Table 3.1: Hotel Table

- **Room table**:

Field Name	Data Type	Key Field	Description
RoomNo	Int	Primary Key	Stores the room number
HotelNo	Char		Stores hotel number

Type	Char		Stores room type

Table 3.2: Room Table

- **Tourist table**:

Field Name	Data Type	Key Field	Description
TouristId	Char	Primary Key	Stores tourist id
TouristName	Char		Stores tourist name
TouristAddress	Char		Stores address
ContactNo	Int		Stores contact number of the tourist

Table 3.3: Tourist Table

- **Booking table**:

Field Name	Data Type	Key Field	Description
HotelNo	Int	Foreign Key	Stores hotel number
TouristId	Char	Foreign Key	Stores tourist id
DateFrom	Datetime		Stores date on which to book the room
DateTo	Datetime		Stores upto what date the room is to be booked
RoomNo	Int	Foreign Key	Stores room number
TouristName	Char		Stores tourist name
BookingNo	Int		Stores booking number of the tourist

Table 3.4: Booking Table

Here, TouristId is a foreign key from Tourist table and (HotelNo, RoomNo) is a foreign key from Room table.

1. Pan World Tours and Travels wants to display all the hotels along with the names of the cities where the hotels are situated. Create a clustered index IX_Hotels on the HotelNo column in the Hotel table.

2. Pan World Tours and Travels wants to keep track of the all the bookings made in a particular hotel. Create a clustered index on IX_Booking on the BookingNo column in the Booking table.

3. The company wants to search for a particular room in a hotel. To search for a particular room, create a composite index IX_Room for the columns RoomNo and HotelNo in the Room table.

Maintaining Indexes

Sr. No.	Assignment Question
1	**RiverPlate University** is an accredited European university, which offers a wide range of courses to its students. It helps the students to receive the very best in terms of education and course content.

Now, the university management is introducing Class Assignment System software, which is an add-on to the traditional Assignment Control System. This allows assigning and monitoring the student-assignment-department details on a class-by-class basis. The software controls and provides accurate, real-time information from a central server and database to all of the educators and constituents responsible for success of the students.

Hence, to create such an application, a database is required which stores details of assignments undertaken by students. The database should have the following tables:

- **Student Table**:

Field Name	Data Type	Key Field	Description
StudentNo	Int	Primary Key	Stores student number
StudentName	Char (30)		Stores student name
StudentAddress	Varchar(Max)		Stores address of the student
PhoneNo	Int		Stores phone number of the student

Table 4.1: Student Table

- **Department Table**:

Field Name	Data Type	Key Field	Description
DeptNo	Int	Primary Key	Stores department number
DeptName	Char (30)		Stores department name
DeptManagerNo	Int		Stores department manager number
ManagerName	Char(30)		Stores manager name

Table 4.2: Department Table

- **Assignment table**:

Field Name	Data Type	Key Field	Description
AssignmentNo	Int	Primary Key	Stores assignment number

AssignmentName	Char (30)		Stores assignment name
Description	Varchar(Max)		Stores description
AssignmentManagerNo	Int		Stores manager number

Table 4.3: Assignment Table

- **Works_Assign table**:

Field Name	Data Type	Key Field	Description
JobID	Int	Primary Key	Stores job id
StudentNo	Int		Stores student number
AssignmentNo	Int		Stores assignment number
TotalHours	Int		Stores total hours allotted
JobDetails	XML		Stores the details of the work assigned

Table 4.4: Works_Assign Table

Here, in this table, JobID is specified as primary key. StudentNo is a foreign key from the Student table and AssignmentNo is a foreign key from the Assignment table.

1. The management of the RiverPlate University wants to display the name of the students and their student number. Create a clustered index IX_Student for the StudentNo column in the Student table, so that while the index is being created, the tables and the indexes can be used for queries and data modification.

2. Alter and rebuild the index IX_Student created on the Student table, so that the tables and indexes cannot be used for queries and data modification.

3. The Management at the RiverPlate University wants to retrieve the name of the Department, department manager and the department number. Create a nonclustered index IX_Dept on the Department table using the key column DeptNo and two non-key columns DeptName and DeptManagerNo.

4. Create a partitioned index named IX_Assign on the Assignment table using the PS_Assignment_Details partition scheme.

5. The University wants to retrieve the assignments which are assigned to the students. Create a primary XML index PXML_Works on the JobID column of the Works_Assign table.

Implementing Views

Sr. No.	Assignment Question
1	Cosmos Electronics Ltd. employs more than 1000 workers in its units. Some of these are at junior level while some are at senior level depending upon their expertise and years of experience. Each employee is given annual leave based on the designation. The management at Cosmos Electronics Ltd. is planning to computerize their human resources department and all the data pertaining to employees will now be stored in SQL Server 2005 databases. Two of the most vital tables in the Employees database are shown below:

- **EmpDetails Table**:

Field Name	Data Type	Key Field	Description
Emp_Id	varchar(5)	Primary Key	Stores employee identification number
FirstName	varchar(30)		Stores first name of the employee
LastName	varchar(30)		Stores last name of the employee
Address	varchar(60)		Stores address of the employee
PhoneNumber	varchar(20)		Phone number of the employee, it could be landline or mobile
Dept_Id	varchar(4)		Stores department id of the department to which the employee belongs
Designation	varchar(30)		Stores designation or job role of the employee
Salary	money		Stores salary of the employee
Join_date	datetime		Stores date of joining for the employee
Performance_Rating	int		Stores Rating of the employee

Table 5.1: EmpDetails Table

- **LeaveDetails Table**:

Field Name	Data Type	Key Field	Description
Emp_Id	varchar(5)	Primary Key	Stores employee identification number
LeaveTaken	int		Stores the number of leaves taken by the employee
FromDate	datetime		Date when the leave started
ToDate	datetime		Date upto which leave was taken

Reason	xml		Reason for the leave

Table 5.2: LeaveDetails Table

1. Using SQL Server 2005 and Transact SQL statements, create the above tables in a database named **Employees**. Add at least 5 records to the tables.

2. Cosmos Electronics Ltd. has decided to allow the employees to login to the database management system and view information. However, at the same time, Cosmos needs to protect certain sensitive and confidential data such as salary, address and phone number of all employees from being viewed. To enable the employees to be able to see specific information, without displaying the confidential information, a view needs to be created.

3. Create a view named **Emp_Public_info** based on information from the tables **Emp_Details** and **Leave_Details** such that the view definition itself cannot be viewed at any particular point of time. The information that is to be included in the view is given below: Emp_ID, FirstName, LastName, Department, Designation, Join_Date, LeaveTaken, FromDate, ToDate, Reason. Note that the reason for leave will be stored in XML format.

4. Test the view by displaying information from it. Display all the records in the view. Display only the top 3 records in the view alphabetically sorted by FirstName.

5. Change the view such that the Join_Date column is no longer visible in the view.

6. Finally, assuming that the view is not proving useful and many employees have not used it, it has been decided by Cosmos to remove the view. Write the statements to remove the view.

7. Write statements to check if the original tablets still exist or have been deleted upon deletion of the view.

Managing Views

Sr. No. **Assignment Question**

1 BookParadise is an online library management system used by a library in Seattle. The software makes use of SQL Server 2005 databases. Information about thousands of books are maintained and updated regularly. In the recent few years, BookParadise has grown in size after receiving international funding and the number of books has increased tremendously. This increase in the number of books has made searching for books very difficult. Also, BookParadise needs to now allow searches to be made by users displaying only selective information to them. The entire content of the tables are not to be displayed to the users. Towards this end, views will be created to enable the readers to display information about books, in a fast and efficient manner.

The **Books** table in the **BookParadise** database has the following structure:

- **Books**:

Field Name	Data Type	Key Field	Description
BookCode	varchar(5)	Primary Key	Book Identification Code
Title	varchar(30)		Title of the book
Author	varchar(30)		Author of the book
Edition	int		Edition number
RatePurchased	money		Cost price of the book
PurchaseDate	datetime		Date when book was purchased
VendorName	varchar(30)		Vendor who sold the book
BookStatus	varchar(15)		Indicates whether book is available or not

Table 6.1: Books Table

All the above fields, except the primary key, may accept null values.

1. Using SQL Server 2005 and Transact SQL statements, create the above table in a database named BookParadise. Add at least seven records to the table.

2. Next, create an indexed view named BookInfo on the table which will contain columns BookCode, Title, Author, Edition, and BookStatus. This view will need to check for domain integrity. Use appropriate options to ensure this.

3. Test the view by displaying information from it. Display all the records in the view. Also, display the top 3 records in the view alphabetically sorted by the column Author.

4. Add three more records to the view.

5. Assuming that there is an author named Mary Clark whose books are listed in the BookInfo view, write the statements to replace all occurrences of Mary Clark in the column Author with Mary Higgins Clark.

6. Remove all the books from the view whose edition is less than 2.

7. Finally, write the statements to remove the view.

Introducing Stored Procedures

Sr. No. **Assignment Question**

1 ToyzUnlimited is a trendy toy store based in California. It buys toys from manufacturers, stocks them in its store and sells them for profits. ToyzUnlimited maintains the details of all branded toy products in a SQL Server 2005 database. To speed up the day-to-day tasks and operations related to the database, it has been decided to use SQL Server 2005 stored procedures for commonly performed tasks.

➢ **Toys**:

Field Name	Data Type	Key Field	Description
ProductCode	varchar(5)	Primary Key	Product Code that uniquely identifies each toy
Name	varchar(30)		Name of the toy
Category	varchar(30)		Category of the toy, such as for example, block building, board games, puzzles etc
Manufacturer	varchar(40)		Manufacturer name
AgeRange	varchar(15)		Age range for the kids to use the toy. Eg: 3-5 years
UnitPrice	money		Price of the toy in dollars
Netweight	int		Weight of the toy in grams
QtyOnHand	int		Quantity available

Table 7.1: Toys Table

1. Start with creating the table Toys. The structure of the table is described above. Add at least 5 records to the table. Ensure that the value of the column QtyOnHand is more than 20 for each of the toys.

2. Write statements to create a stored procedure named HeavyToys that will list names of all the toys that are above 500 grams.

3. Write statements to create a stored procedure named PriceIncrease that will increment the unitprice of all toys by 10 dollars.

4. Write statements to create a stored procedure QtyOnHand that will decrease the quantity on hand of all toys by 5.

5. Execute the stored procedures HeavyToys, PriceIncrease and QtyOnHand.

6. In SQL Server Management Studio, locate the extended stored procedures defined under the master database and execute the following procedures in a query window:

```
                    sys.xp_getnetname
                    sys.xp_fixeddrives
                    sys.xp_loginconfig
```

More About Stored Procedures

Sr. No.	Assignment Question
1	ToyzUnlimited maintains the details of all branded toy products in SQL Server 2005 databases and is now using stored procedures for day-to-day tasks. 1. Assuming that the stored procedures named PriceIncrease, QtyOnHand and HeavyToys have been created, now write statements to view the definition of the stored procedures using all these 3 approaches one by one: ➢ `sp_helptext` system stored procedure ➢ `sys.sql_modules` system view ➢ `OBJECT_DEFINITION` function 2. Write statements to display the dependencies for each of the 3 stored procedures. 3. Change the procedures PriceIncrease and QtyOnHand such that they will also display the newly updated values of price and quantity after the updation of these columns have taken place through the stored procedures mentioned above. 4. Write statements to create a stored procedure named SpecificPriceIncrease that will increment the unitprice of given toys by a given amount. 5. Modify the SpecificPriceIncrease stored procedure such that it additionally returns the number of rows updated. 6. Modify the SpecificPriceIncrease stored procedure such that it calls the HeavyToys stored procedure within it. 7. Implement error-handling mechanism for all the stored procedures created so far. 8. Remove all the stored procedures that were created so far.

Introduction to Triggers

Sr. No. **Assignment Question**

1 Euro Airlines is a newly launched airline service that operates flights to and from various cities all over Europe. The company maintains the data pertaining to day-to-day transactions regarding flight services in SQL Server 2005 databases. To enable efficient and faster performance, Euro Airlines has decided to incorporate use of triggers in their database applications. The detailed list of operations to be performed are listed below:

1. Start with the creation of the tables in a database named EuroAirlines. The structure of each of the tables is given herewith:

- **Flight**:

Field Name	Data Type	Key Field	Description
Aircraft_code	varchar(10)	Primary key	Stores aircraft code
Type	varchar(6)		Describes the type of aircraft
Source	varchar(20)		Stores the name of the city from where the aircraft will depart
Destination	varchar(20)		Stores the name of the city where the aircraft will arrive
Dep_time	varchar(10)		Stores departure time
Journey_hrs	int		Stores journey hours

Table 9.1: Flight Table

- **Flight_Details**:

Field Name	Data Type	Key Field	Description
Aircraft_code	varchar(10)	Primary Key	Stores aircraft code
Class_code	varchar(10)		Stores the class, whether first, business or economy
Fare	money		Stores the fare amount
Seats	int		Stores total number of seats on the flight

Table 9.2: Flight_Details Table

2. Write statements to create a trigger CheckSeats that will activate whenever a new row is being inserted into the Flight_Details table. The maximum limit of seats that a flight can contain is 150. The trigger should check for the value of seats being inserted. If it is more than 150, the INSERT operation is not allowed to succeed.

3. Insert at least five records in each table.

4. Write statements to create a trigger UpdateValid that will activate whenever a row is being updated in the Flight_Details table. The trigger should determine if the Seats column is present in the list of columns being updated. If yes, the UPDATE operation should not succeed because the Seats column is defined as a constant and cannot be changed.

5. Implement a cascade delete by using a trigger Casc_Delete such that if a particular Aircraft_code is deleted from the Flight table, the corresponding row from Flight_details table is also deleted.

6. Write statements to display the list of triggers created in the EuroAirlines database.

7. Write statements to view the definitions of triggers defined in the EuroAirlines database.

8. Write statements to create a DDL trigger ProhibitDelete that will activate whenever a user is trying to delete a table from the EuroAirlines database. The trigger must not allow a user to perform deletes and must display a message “You are not allowed to delete tables in this database”.

--- End of Assignments ---