**MS SQL Practice Exam**

**Duration:** 60 minutes  
**Total Marks:** 50

**Section A: Storing and Manipulating Data (15 Marks)**

**1. Storing Data in a Table (3 Marks)**  
a) Write an SQL query to create a table named **Employees** with the following fields:

* EmployeeID (Primary Key, INT, Auto-increment)
* Name (VARCHAR(100), NOT NULL)
* Age (INT)
* Department (VARCHAR(50))
* Salary (DECIMAL(10,2))

Create table Employees(EmployeeId int primary key identity(1,1), Name varchar(100) not null, Age int, Department varchar(50), Salary decimal(10,2));

b) Insert three records into the **Employees** table.

Insert into Employees(’Ajay’,25,’HR’,30000);

Insert into Employees(’Balaji’,27,’Finance’,39000);

Insert into Employees(’Charan’,24,’IT’,36000);

**2. Updating Data in a Table (3 Marks)**  
Write an SQL query to update the **Salary** of employees in the **HR** department by 10%.

Update Employees Set Salary=Salary\*1.10 where Department =’HR’;

**3. Deleting Data from a Table (3 Marks)**  
Write an SQL query to delete all employees from the **IT** department.

Delete from Employees where Department=’IT’;

**4. Demo: Manipulating Data in Tables (6 Marks)**  
a) Insert a new employee into the **Employees** table, but ensure the **Salary** is at least 30000. (2 Marks)

Insert into Employees(’Bala’,67,’Finance’,55000);  
b) Update the **Department** of employees who earn more than 50000 to **Senior Staff**. (2 Marks)

Update Employees set Department=’Senior Staff’ where Salary>50000;  
c) Delete employees older than 60 years from the table. (2 Marks)

Delete from Employees where Age>60;

**Section B: Retrieving and Filtering Data (35 Marks)**

**5. Retrieving Specific Attributes (3 Marks)**  
Write an SQL query to retrieve only the **Name** and **Salary** of all employees.

Select Name,Salary from Employees;

**6. Retrieving Selected Rows (3 Marks)**  
Write an SQL query to retrieve employees from the **HR** department who have a salary greater than 50,000.

Select \* from Employees where Department=’HR’ and Salary>50000;

**7. Demo: Retrieving Data (4 Marks)**  
Write an SQL query to retrieve all employees sorted by **Salary** in descending order.

Select \* from Employees order by Salary desc;

**8. Filtering Data - WHERE Clauses (5 Marks)**  
a) Write an SQL query to retrieve employees whose **Age** is greater than 30. (2 Marks)

Select \* from Employees where Age>30;  
b) Retrieve employees whose **Department** is either **HR** or **Finance**. (3 Marks)

Select \* from Employees where Department=’HR’ or Department=’Finance’;

**9. Filtering Data - Operators (10 Marks)**  
a) Retrieve employees whose **Salary** is between 30,000 and 60,000. (2 Marks)

Select \* from Employees where Salary between 30000 and 60000;  
b) Retrieve employees whose **Name** starts with "A". (2 Marks)

Select \* from Employees where Name like ‘A%’;  
c) Retrieve employees who do NOT belong to the **IT** department. (2 Marks)

Select \* from Employees where Department != ‘IT’;  
d) Retrieve employees whose **Department** is either "Sales" or "Marketing" using the **IN** operator. (2 Marks)

Select \* from Employees where Department in (‘Sales’,‘Marketing’);   
e) Retrieve employees with distinct **Department** names. (2 Marks)

Select distinct Department from Employees;

**10. Column & Table Aliases (3 Marks)**  
Write an SQL query that retrieves **EmployeeID, Name, and Salary**, renaming **EmployeeID** as "ID" and **Salary** as "Monthly Income".

Select EmployeeID as ID, Name, Salary as Monthly\_Income from Employees;

**11. Demo: Filtering Data (4 Marks)**  
Write an SQL query to retrieve employees whose **Name** contains "John" and whose salary is greater than 40,000.

Select \* from Employees where Name like ‘%John%’ and Salary>40000;