SQL Hands On 5

Instruction:

1. Create a stored procedure for employee table.

- select all employee records.

```
Export of routine `GetAll`

1  DELIMITER $$
2  CREATE DEFINER=`root`@`localhost` PROCEDURE `GetAll`()
3  SELECT * FROM test_employee$$
4  DELIMITER ;
```

- select employee by id.(With Parameter)

```
Export of routine `GetEmployeeByID`

1  DELIMITER $$
2  CREATE DEFINER=`root`@`localhost` PROCEDURE
   `GetEmployeeByID`(IN `id` INT(11))
3  SELECT * FROM test_employee WHERE EmpNo = id$$
4  DELIMITER ;
```

- insert new employee.(With Parameter)

```
Export of routine `AddEmployee`

1  DELIMITER $$
2  CREATE DEFINER=`root`@`localhost` PROCEDURE `AddEmployee`(IN
   `empID` INT(11), IN `name` VARCHAR(100), IN `job`
   VARCHAR(100), IN `mgr` INT(11), IN `datehired` DATETIME, IN
   `salary` DECIMAL, IN `comission` INT(11), IN `deptID`
   INT(11))
3  INSERT INTO `test_employee`(`EmpNo`, `EmpName`, `Job`,
   `Mgr`, `HireDate`, `Salary`, `Commission`, `DeptID`) VALUES
   (empID,name,job,mgr,datehired,salary,comission,deptID)$$
4  DELIMITER ;
```

- update employee records.(With Parameter)

Export of routine `UpdateEmployee`

```
1 DELIMITER $$
2 CREATE DEFINER=`root`@`localhost` PROCEDURE
  `UpdateEmployee`(IN `empID` INT(11), IN `newEmpID` INT(11),
  IN `name` VARCHAR(100), IN `job` VARCHAR(100), IN `mgr`
  INT(11), IN `datehired` DATETIME, IN `salary` DECIMAL, IN
  `comission` INT(11), IN `deptID` INT(11))
3 UPDATE `test_employee` SET
  `EmpNo`=newEmpID,`EmpName`=name,`Job`=job,`Mgr`=mgr,`HireDat
  e`=datehired,`Salary`=salary,`Commission`=comission,`DeptID`
  =deptID WHERE `EmpNo` = empID$$
4 DELIMITER ;
```

- delete employee.(With Parameter)

Export of routine `DeleteEmployee`

```
1 DELIMITER $$
2 CREATE DEFINER=`root`@`localhost` PROCEDURE
  `DeleteEmployee`(IN `id` INT(11))
3 DELETE FROM `test_employee` WHERE EmpNo = id$$
4 DELIMITER ;
```

2. Create a stored procedure for department table
- select all department records.

Export of routine `GetAllDepartment`

```
1 DELIMITER $$
2 CREATE DEFINER=`root`@`localhost` PROCEDURE
  `GetAllDepartment`()
3 SELECT * FROM test_department$$
4 DELIMITER ;
```

- select employee by department name.(With Parameter)

Export of routine `GetDepartmentByName`

```
1 DELIMITER $$
2 CREATE DEFINER=`root`@`localhost` PROCEDURE
  `GetDepartmentByName`(IN `name` VARCHAR(100))
3 SELECT * FROM `test_department` WHERE DeptName = name$$
4 DELIMITER ;
```

■ insert new department.(With Parameter)

Export of routine `AddDepartment`

```
1 DELIMITER $$
2 CREATE DEFINER=`root`@`localhost` PROCEDURE
  `AddDepartment`(IN `id` INT(11), IN `name` VARCHAR(100), IN
  `location` VARCHAR(100))
3 INSERT INTO `test_department`(`DeptID`, `DeptName`,
  `Location`) VALUES (id,name,location)$$
4 DELIMITER ;
```

■ update department records.(With Parameter)

Export of routine `UpdateDepartment`

```
1 DELIMITER $$
2 CREATE DEFINER=`root`@`localhost` PROCEDURE
  `UpdateDepartment`(IN `id` INT(11), IN `newID` INT, IN
  `name` VARCHAR(100), IN `location` VARCHAR(100))
3 UPDATE `test_department` SET
  `DeptID`=newID,`DeptName`=name,`Location`=location WHERE
  DeptID = id$$
4 DELIMITER ;
```

■ delete department.(With Parameter)

Export of routine `DeleteDepartment`

```
1 DELIMITER $$
2 CREATE DEFINER=`root`@`localhost` PROCEDURE
  `DeleteDepartment`(IN `id` INT(11))
3 DELETE FROM `test_department` WHERE DeptID = id$$
4 DELIMITER ;
```

3. Create a stored procedure for the following requirement.

- get the list of employees by searching department name.(With Parameter)

Export of routine `GetEmployeeByDeptName`

```
1 DELIMITER $$
2 CREATE DEFINER=`root`@`localhost` PROCEDURE
  `GetEmployeeByDeptName`(IN `deptName` VARCHAR(100))
3 SELECT *
4 FROM `test_employee`
5 INNER JOIN test_department ON test_employee.DeptID =
  test_department.DeptID
6 WHERE  test_department.DeptName = deptName$$
7 DELIMITER ;
```

- get the list of distinct salaries per department name.

Export of routine `GetDistinctSalary`

```
1 DELIMITER $$
2 CREATE DEFINER=`root`@`localhost` PROCEDURE
  `GetDistinctSalary`()
3 SELECT DISTINCT e.Salary, d.DeptName
4 FROM test_employee e
5 INNER JOIN test_department d ON e.DeptID = d.DeptID
6 ORDER BY DeptName$$
7 DELIMITER ;
```

- get the name of employee and their assigned location. (With Parameter)

Export of routine `GetLocationByEmpName`

```
1 DELIMITER $$
2 CREATE DEFINER=`root`@`localhost` PROCEDURE
  `GetLocationByEmpName`(IN `name` VARCHAR(100))
3 SELECT e.EmpName, d.Location
4 FROM test_employee e
5 INNER JOIN test_department d ON e.DeptID = d.DeptID
6 WHERE e.EmpName = name$$
7 DELIMITER ;
```