VIEW

a. *Create view (SSN, Full Name of employee, Project Number, Project Name, Hours) that includes information about employees, projects, and hours for those projects at Houston.*

```
CREATE VIEW EmployeeProjectHouston AS
SELECT e.SSN,
        CONCAT(e.FirstName, ' ', e.LastName) AS FullName,
        p.ProjectNumber, p.ProjectName, w.Hours
FROM Employee e
JOIN WorksOn w ON e.SSN = w.EmployeeSSN
JOIN Project p ON w.ProjectNumber = p.ProjectNumber
WHERE p.Location = 'Houston';
```

b. *Create view (SSN, Full Name of employee, Number of dependent) that includes information about employees who have the number of dependents greater than 2.*

```
CREATE VIEW EmployeeDependents AS
SELECT e.SSN,
        CONCAT(e.FirstName, ' ', e.LastName) AS FullName,
        COUNT(d.DependentName) AS NumberOfDependents
FROM Employee e
JOIN Dependent d ON e.SSN = d.EmployeeSSN
GROUP BY e.SSN, e.FirstName, e.LastName
HAVING COUNT(d.DependentName) > 2;
```

c. *Create view (Full Name of employee, date of birth, gender) for those employees who have their birthdate in July. Make this view as read-only.*

```
CREATE VIEW EmployeeJulyBirthdate AS
 SELECT CONCAT(e.FirstName, ' ', e.LastName) AS FullName,
        e.BirthDate,
        e.Gender,
FORM Employee e
WHERE
        MONTH(e.BirthDate) = 7
WITH CHECK OPTION
```

TRIGGER, FUNTION, STORE PROCEDURE, CURSOR

a. *Employees in the company must be older than 18 years old. Write a trigger to implement this constraint.*

```
DELIMITER $$
CREATE TRIGGER CheckEmployeeAge
BEFORE INSERT ON Employee
FOR EACH ROW
BEGIN
        IF TIMESTAMPDIFF(YEAR, NEW.BirthDate, CURDATE()) < 18 THEN
                SIGNAL SQLSTATE '45000'
```

```
            SET MESSAGE_TEXT = 'Employee must be at least 18 years old.';
        END IF;
END$$
DELIMITER ;
```

b. *Write a function that returns the total number of projects when given an employee's ID.*
Input: employee
ID Output: total number of projects

```
DELIMITER $$
CREATE FUNCTION GetTotalProjects(empID INT)
RETURNS INT
DETERMINISTIC
BEGIN
        DECLARE totalProjects INT;
        SELECT COUNT(*) INTO totalProjects
        FROM WorksOn
        WHERE EmployeeSSN = empID;
        RETURN totalProjects;
END$$
DELIMITER ;
```

c. *Create a store procedure that prints SSN, Full name of employee, Department name, and annual salary.*

```
DELIMITER $$
CREATE PROCEDURE PrintEmployeeInfo()
BEGIN
        SELECT e.SSN,
            CONCAT(e.FirstName, ' ', e.LastName) AS FullName,
            d.DepartmentName, (e.Salary * 12) AS AnnualSalary
        FROM Employee e
        JOIN Department d ON e.DepartmentID = d.DepartmentID;
END$$
DELIMITER ;
```