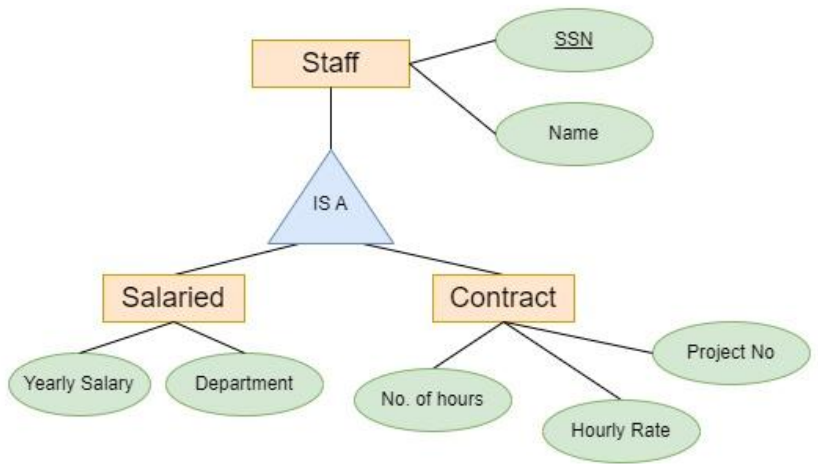
PART 1

Solution 1:

Adhiraj Budukh
abudukh@wpi.edu



Schemas :

#Table 1:

```
CREATE TABLE staff (
    SSN PRIMARY KEY,
    Name,
    type,
    Yearly_salary,
    Department,
    Contract_Hours,
    Hourly_Rate,
    ProjectNum
);
```

#Table 2:

```
CREATE TABLE staff (
  SSN PRIMARY KEY,
  Name,
  Contract_Hours,
  Hourly_Rate,
  ProjectNum
);


CREATE TABLE salaried_employee (
  SSN PRIMARY KEY,
  Yearly_Salary,
  Department
);

ALTER TABLE salaried_employee ADD FOREIGN KEY (ssn) REFERENCES staff (ssn);
```

#Table 3:

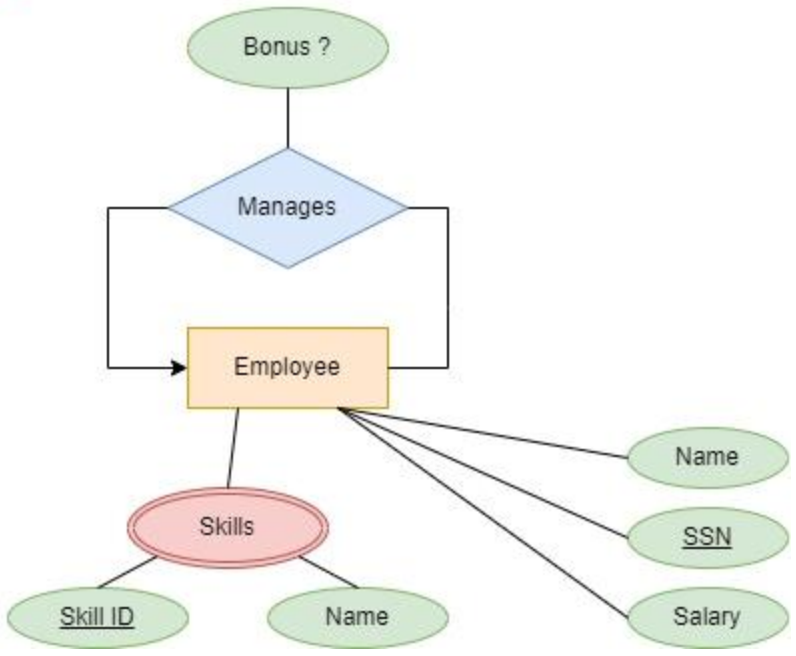
```
CREATE TABLE staff (
   SSN PRIMARY KEY,
   Name
);

CREATE TABLE salaried_employee (
   SSN PRIMARY KEY,
   Yearly_Salary,
   Department,
   FOREIGN KEY (SSN) REFERENCES staff(SSN)
);

CREATE TABLE contract_employee (
   SSN PRIMARY KEY,
   Contract_Hours,
   Hourly_Rate,
   ProjectNum,
   FOREIGN KEY (SSN) REFERENCES staff(SSN)
);
```

- The none-overlap constraint can also be enforced by adding the 'type' column in the staff table. This ensures that no two employees can have the same type.
- Adding primary keys, foreign keys can also be enforced in all three designs.
- The choice of which design is best will depend on the specific requirements of the application. If the application needs to track a lot of data about salaried employees, then Design 1 might be the best choice. If the application needs to track a lot of data about contract employees, then Design 2 might be the best choice. If the application needs to track a balanced amount of data about both salaried employees and contract employees, then Design 3 might be the best choice.
- The none-overlap constraint can also be enforced by setting the NOTNULL constraint on the type column in the staff table. This ensures that employees cannot have more than one type.

PART 2

1.

Part 2

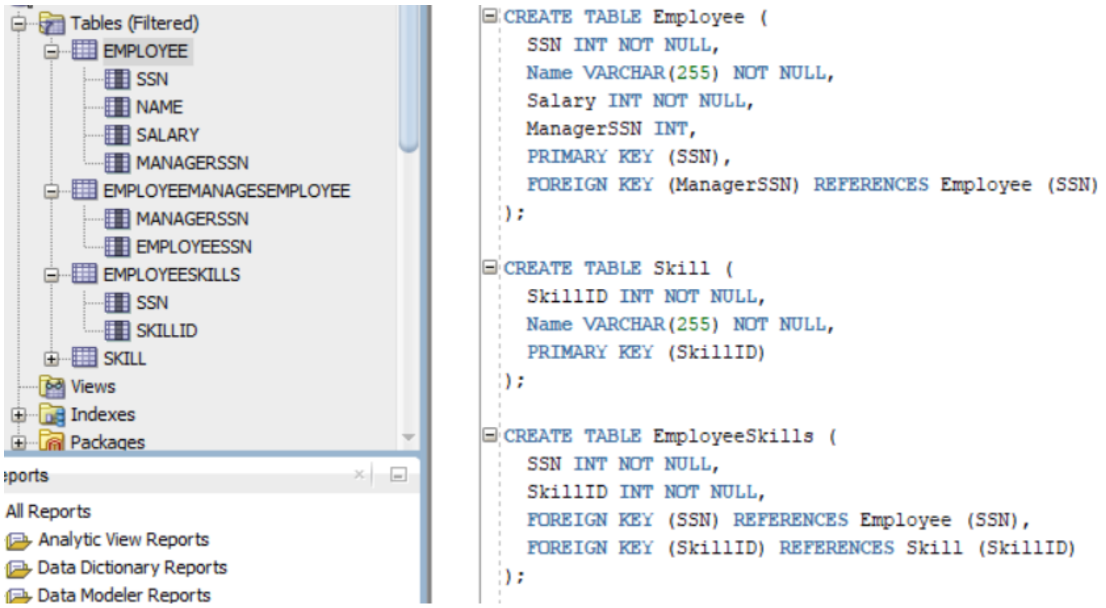


2. Schemas :

```
CREATE TABLE Employee (
  SSN INT NOT NULL,
  Name VARCHAR(255) NOT NULL,
  Salary INT NOT NULL,
  ManagerSSN INT,
  PRIMARY KEY (SSN),
  FOREIGN KEY (ManagerSSN) REFERENCES Employee (SSN)
);
```

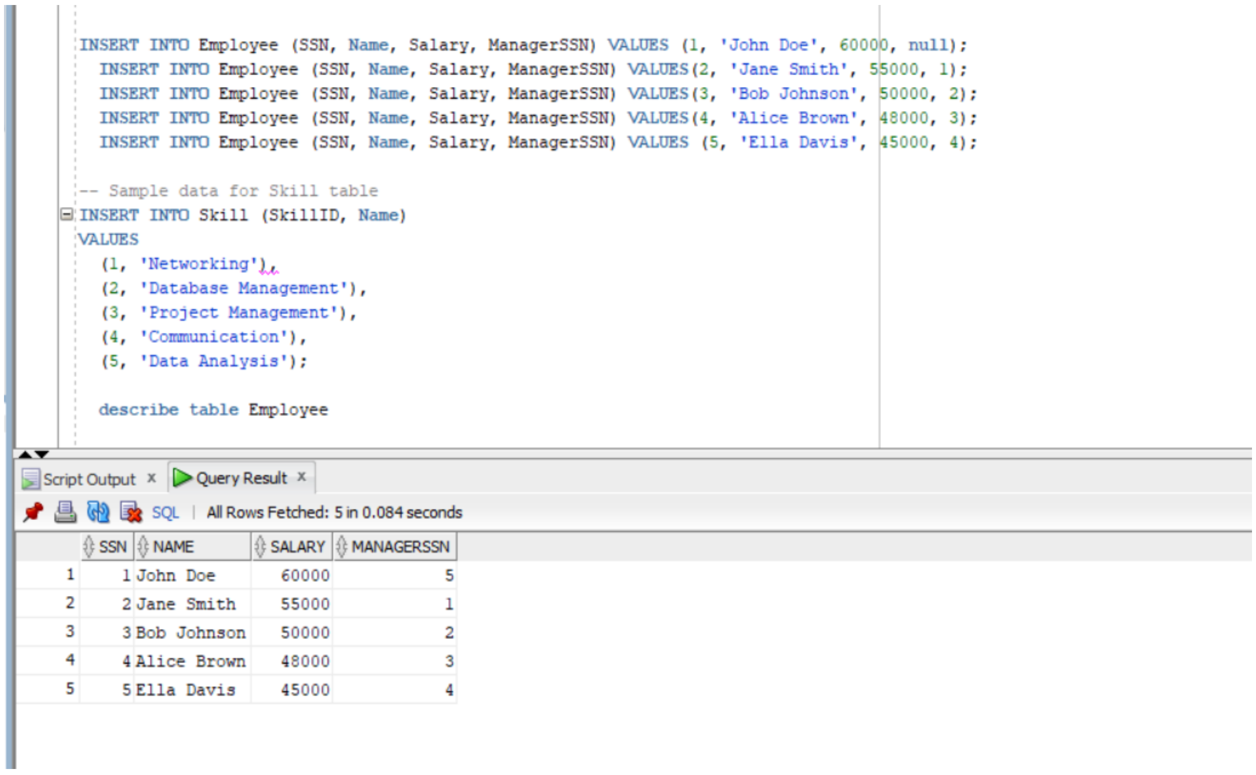
```
CREATE TABLE Skill (
  SkillID INT NOT NULL,
  Name VARCHAR(255) NOT NULL,
  PRIMARY KEY (SkillID)
);

CREATE TABLE EmployeeSkills (
  SSN INT NOT NULL,
  SkillID INT NOT NULL,
  FOREIGN KEY (SSN) REFERENCES Employee (SSN),
  FOREIGN KEY (SkillID) REFERENCES Skill (SkillID)
);
```

3. Running Schemas in the oracle:



```
CREATE TABLE Employee (
   SSN INT NOT NULL,
   Name VARCHAR(255) NOT NULL,
   Salary INT NOT NULL,
   ManagerSSN INT,
   PRIMARY KEY (SSN),
   FOREIGN KEY (ManagerSSN) REFERENCES Employee (SSN)
);

CREATE TABLE Skill (
   SkillID INT NOT NULL,
   Name VARCHAR(255) NOT NULL,
   PRIMARY KEY (SkillID)
);

CREATE TABLE EmployeeSkills (
   SSN INT NOT NULL,
   SkillID INT NOT NULL,
   FOREIGN KEY (SSN) REFERENCES Employee (SSN),
   FOREIGN KEY (SkillID) REFERENCES Skill (SkillID)
);
```

Creating dataset and insert values



```
INSERT INTO Employee (SSN, Name, Salary, ManagerSSN) VALUES (1, 'John Doe', 60000, null);
  INSERT INTO Employee (SSN, Name, Salary, ManagerSSN) VALUES(2, 'Jane Smith', 55000, 1);
  INSERT INTO Employee (SSN, Name, Salary, ManagerSSN) VALUES(3, 'Bob Johnson', 50000, 2);
  INSERT INTO Employee (SSN, Name, Salary, ManagerSSN) VALUES(4, 'Alice Brown', 48000, 3);
  INSERT INTO Employee (SSN, Name, Salary, ManagerSSN) VALUES (5, 'Ella Davis', 45000, 4);

-- Sample data for Skill table
INSERT INTO Skill (SkillID, Name)
VALUES
  (1, 'Networking'),
  (2, 'Database Management'),
  (3, 'Project Management'),
  (4, 'Communication'),
  (5, 'Data Analysis');

  describe table Employee
```

All Rows Fetched: 5 in 0.084 seconds

```
     SSN NAME        SALARY MANAGERSSN
1      1 John Doe     60000          5
2      2 Jane Smith   55000          1
3      3 Bob Johnson  50000          2
4      4 Alice Brown  48000          3
5      5 Ella Davis   45000          4
```

4. Assumptions :

Employee can not manages itself.

Employees can have multiple skills

Every employee must have manager

5. We cannot utilize bonus options through the schema itself, since we need to write the trigger condition for it. After using trigger feature, it allows us for particular condition we can trigger bonus money gets added into salary.