**Database Final Project**

There is an XYZ Company, which purchases some parts from vendors to produce some products. It has several departments, marketing sites, and parts supply vendors in the company.

1) For each department, department id and department name will be recorded.

2) People in the company can be divided into three types -- employees, customers, and potential employees. Each person can belong to more than one type. Each person in the company has the following attributes: Personal\_ID, Name (Last Name, First Name), Age (below 65), Gender, Address (address line 1, address line 2, city, state, zipcode), and Phone number (one individual may have more than one phone number). For customers, his/her preferred salesmen were recorded in the system. For employees, Rank and Title (e.g. CEO, Principle, Partner, etc.) will be recorded for them.

3) Each employee of the company must have only one direct supervisor, while each supervisor can have several supervisees. One employee can work for one or more departments at different time. But at one time, one employee can only work for one department. The system needs to record start time and end time for each shift among different department for one employee.

4) Each job position’s information is recorded to hire new people. It contains the Job ID, job description, and posted date in the system.

5) The job positions are posted by the departments. Both existing employees and potential employees can apply each job post by any department. The company will select some candidates from the applications and make interviews.

6) For each job position, several interviews will be made to select a suitable person.

7) For each interview, candidates (interviewees), interviewers, job position and interview time are recorded. After each round interview, the interviewers give a grade to it ranging from 0 to 100. The grade over 60 represents that the interviewee pass the interview. One person is selected when her/his average grade is over 70 and she/he passes at least 5 rounds of interviews.

8) For each product in the company, the system needs to record Product ID, Product Type, Size, List Price, Weight, and Style.

9) There are many marketing sites for the company. For each site, Site ID, Site Name, and Site Location are recorded.

10) There are several people working for each site, and meanwhile, one person can work on different sites. It is able to track the details of each sale history --- salesmen, customers, product, sale time, and sites.

11) Part purchase is also a vital activity in the company. The system needs to record each vendor’s Vendor ID, Name, Address, Account Number, Credit Rating, and Purchasing Web Service URL.

12) One vendor may supply many types of parts. The price of the same part type may vary from different vendors but the price of one part type of one vendor will keep same. It is able to track which part types used in each product and the number of each type of part used for the product.

13) In addition, the system maintains the information of each employee’s monthly salary which includes transaction number, pay\_date, and amount (Note: transaction number could be same among different employees. However, for each employee, the transaction number is unique).

**Project Questions**

· Can you think 5 more rules (other than the one explicitly described above) that are likely to be used in a company.

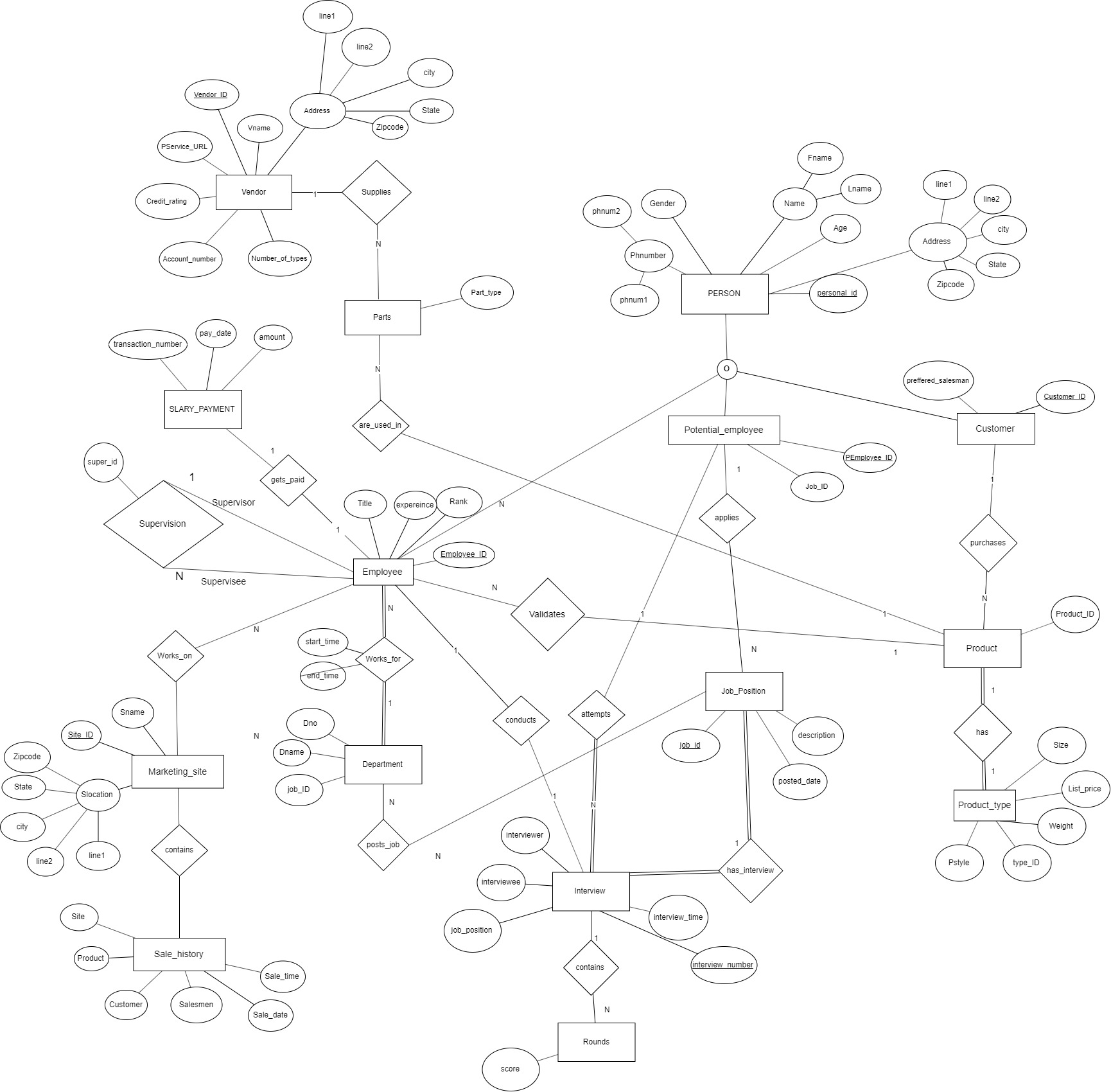
1. Each part needs to have a typeID and serial number to keep track of the part purchases.
2. Each interviewee’s average grade must be recorded
3. Marketing sites must store email of the customer if he/she decides to subscribe.
4. Each product must be tested/validated by the company before selling to the customer.
5. The Supervisor must have at least 3 or more years of experience than the supervisee.
6. The number of rounds interviewee passed must be recorded.

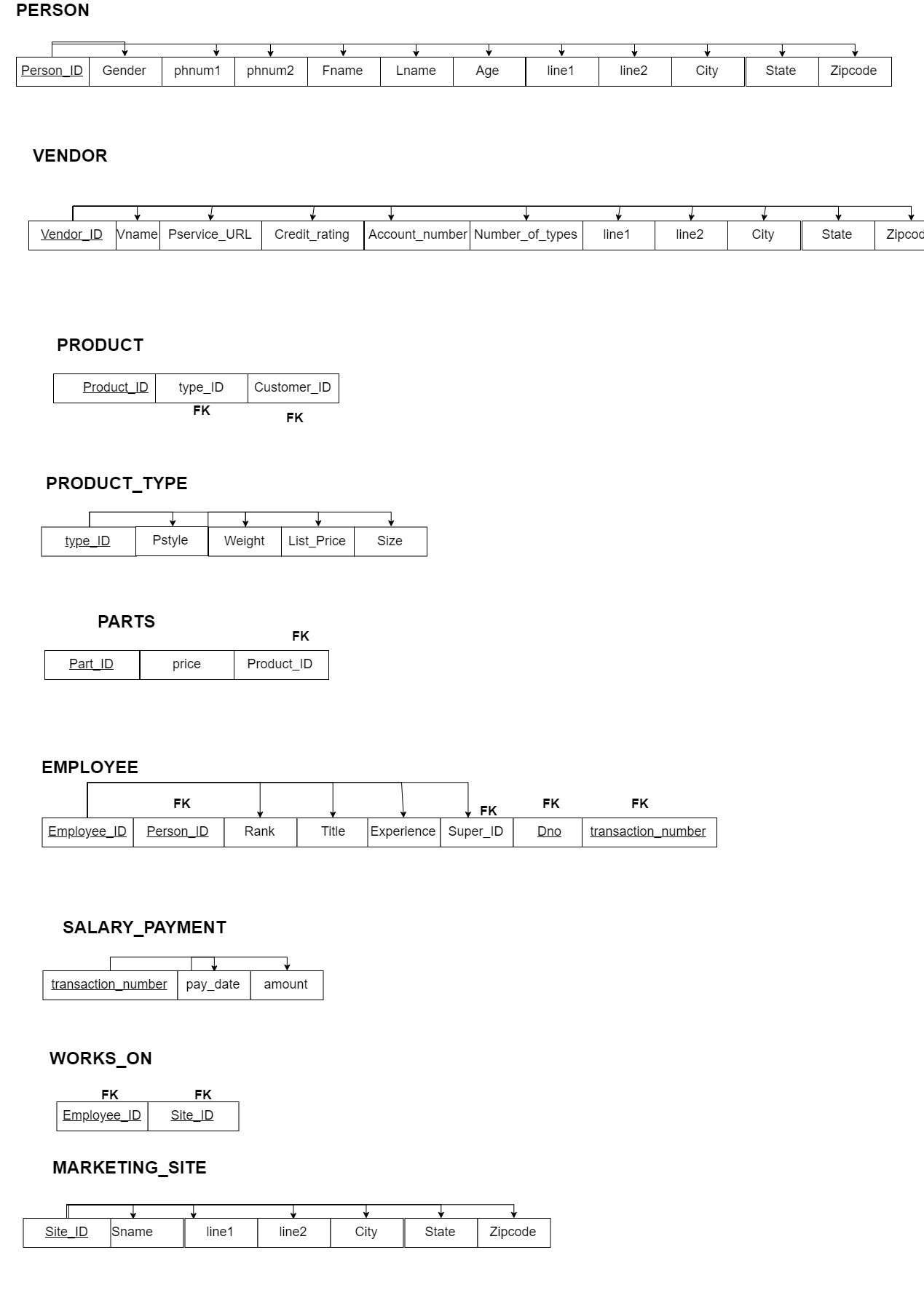
· Is the ability to model super-class/subclass relationships likely to be important in such environment? Why or why not?

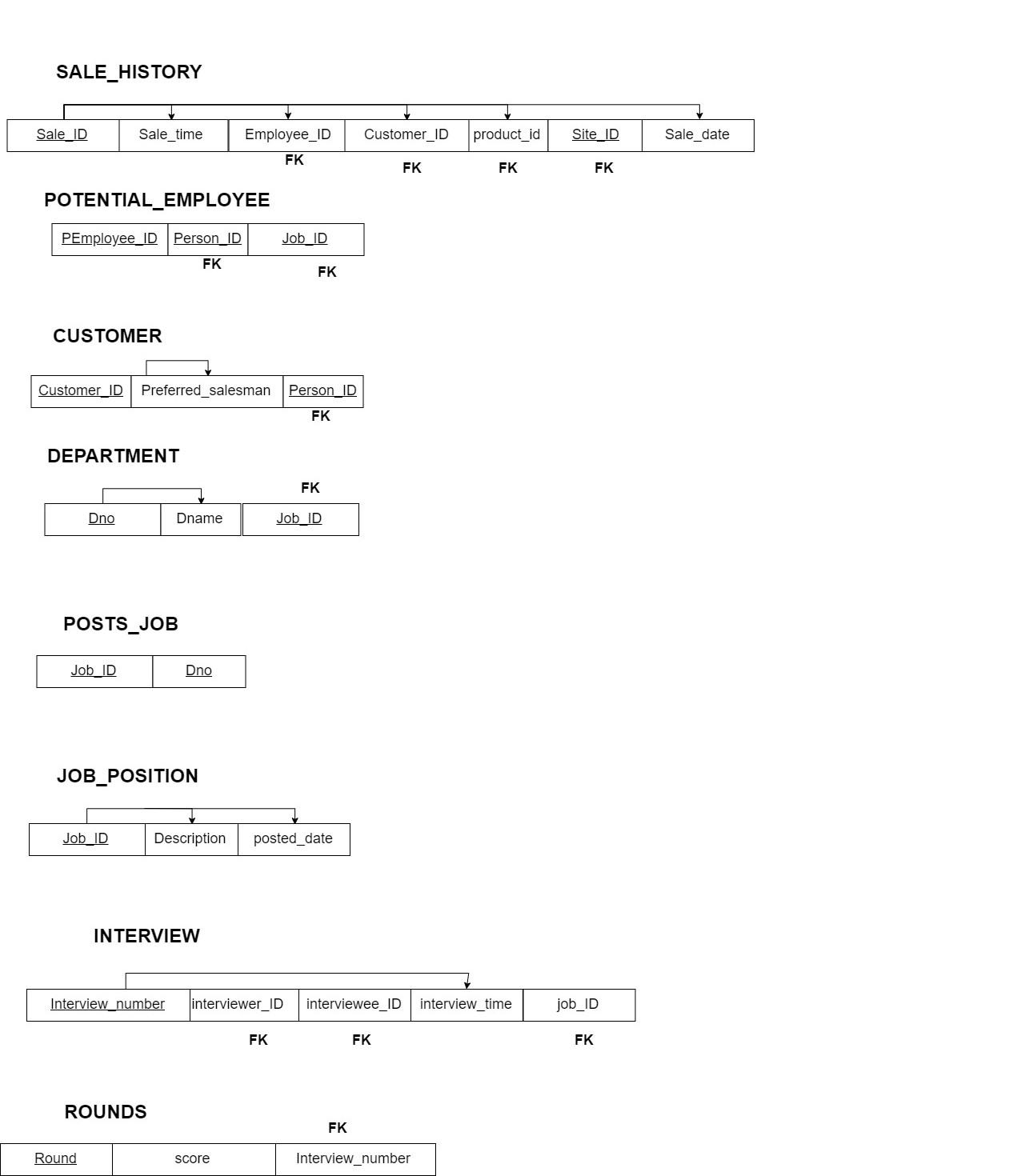
* Modeling a super-class/subclass relationship allows us to share the attributes of the parent with the children. It makes the environment easy to understand and implement for a user.
* It also allows us to represent disjoint or overlapping constraints in the relations. For example, the Person entity has an overlapping constraint on its subclasses – Employee, Potential Employee and Customer. Having the super-class/ subclass relationship helps us understand that a Person in the Company May be 2 or more of its children.

· Justify using a Relational DBMS like Oracle for this project.

Relational database organizes data in a series of table that have interconnectivity amongst them. There are several entities and attributes in this project that require to be stored in different tables that are linked together. By using relational database like Oracle, the tables can be linked using Foreign Keys. Relational database also allows us to handle heavy querying and reading and printing of rows and columns from entities. Because of its readability and easy to use structure, relational database is a better choice to use in this project.

ASSUMPTIONS – If an employee applies, we are assuming that he/she is now a potential employee.





**TABLES CREATION**

CREATE TABLE VENDOR(

Vendor\_ID number(10) NOT NULL,

Vname varchar(30) NOT NULL,

Pservice\_URL varchar(20),

Credit\_rating number(2),

Account\_number number(20),

Number\_of\_types number(3),

line1 varchar(30)NOT NULL,

line2 varchar(30)NOT NULL,

city varchar(30)NOT NULL,

state varchar(30)NOT NULL,

Zipcode number(5)NOT NULL,

PRIMARY KEY(Vendor\_ID)

);

CREATE TABLE PERSON(

Person\_ID number(10) NOT NULL,

Gender char(1),

phnum1 number(10) NOT NULL,

phnum2 number(10),

Fname varchar(30),

Lname varchar(30),

Age number(3),

line1 varchar(30)NOT NULL,

line2 varchar(30)NOT NULL,

city varchar(30)NOT NULL,

state varchar(30)NOT NULL,

Zipcode number(5)NOT NULL,

PRIMARY KEY(Person\_ID)

);

CREATE TABLE PRODUCT\_TYPE(

type\_id number(10) NOT NULL,

Pstyle varchar(20)NOT NULL,

weight number(3)NOT NULL,

List\_price number(10)NOT NULL,

Psize number(3) NOT NULL,

PRIMARY KEY(type\_id)

);

CREATE TABLE CUSTOMER(

customer\_ID number(10) NOT NULL,

Person\_ID number(10) NOT NULL,

Preferred\_salesman varchar(20)NOT NULL,

PRIMARY KEY(customer\_ID),

FOREIGN KEY(Person\_ID) REFERENCES PERSON(Person\_ID)

);

CREATE TABLE PRODUCT(

Product\_ID number(10) NOT NULL,

type\_id number(10) NOT NULL,

customer\_ID number(10) NOT NULL,

PRIMARY KEY(Product\_ID),

FOREIGN KEY(type\_id) REFERENCES PRODUCT\_TYPE(type\_id),

FOREIGN KEY(customer\_ID) REFERENCES CUSTOMER(customer\_ID)

);

CREATE TABLE SALARY\_PAYMENT(

transaction\_number number(10) NOT NULL,

pay\_date date NOT NULL,

amount number(8) NOT NULL,

PRIMARY KEY(transaction\_number)

);

CREATE TABLE DEPARTMENT(

Dno number(10) NOT NULL,

Dname varchar(30) NOT NULL,

PRIMARY KEY(Dno),

);

CREATE TABLE EMPLOYEE(

Employee\_ID number(10) NOT NULL,

Person\_ID number(10) NOT NULL,

Rank number(3) NOT NULL,

Title varchar(20) NOT NULL,

Experience number(2) NOT NULL,

Dno number (10) NOT NULL,

super\_ID number(10),

transaction\_number number(10) NOT NULL,

PRIMARY KEY(Employee\_ID),

FOREIGN KEY(Dno) REFERENCES DEPARTMENT(Dno),

FOREIGN KEY(Person\_ID) REFERENCES PERSON(Person\_ID),

FOREIGN KEY(Employee\_ID) REFERENCES INTERVIEW(Interviewer\_ID),

FOREIGN KEY(Employee\_ID) REFERENCES Potential\_employee(PEmployee\_ID),

FOREIGN KEY(super\_ID) REFERENCES EMPLOYEE(Employee\_ID));

CREATE TABLE SALE\_HISTORY(

Sale\_ID number(10) NOT NULL,

Sale\_time number(4) NOT NULL,

Sale\_date date NOT NULL,

Employee\_ID number(10) NOT NULL,

Customer\_ID number(10) NOT NULL,

Product\_ID number (10) NOT NULL,

Site\_ID number(10) NOT NULL,

PRIMARY KEY(Sale\_ID),

FOREIGN KEY(Employee\_ID) REFERENCES EMPLOYEE(Employee\_ID),

FOREIGN KEY(Customer\_ID) REFERENCES CUSTOMER(customer\_ID),

FOREIGN KEY(Product\_ID) REFERENCES PRODUCT(Product\_ID),

FOREIGN KEY(Site\_ID) REFERENCES MARKETING\_SITE(Site\_ID)

);

CREATE TABLE MARKETING\_SITE(

Site\_ID number(10) NOT NULL,

Sname varchar(30) NOT NULL,

line1 varchar(30)NOT NULL,

line2 varchar(30)NOT NULL,

city varchar(30)NOT NULL,

state varchar(30)NOT NULL,

Zipcode number(5)NOT NULL,

PRIMARY KEY(Site\_ID)

);

CREATE TABLE WORKS\_ON(

Employee\_ID number(10) NOT NULL,

Site\_ID number(10) NOT NULL,

PRIMARY KEY( Employee\_ID, Site\_ID),

FOREIGN KEY(Employee\_ID) REFERENCES EMPLOYEE(Employee\_ID),

FOREIGN KEY(Site\_ID) REFERENCES MARKETING\_SITE(Site\_ID)

);

CREATE TABLE POTENTIAL\_EMPLOYEE(

PEmployee\_ID number(10) NOT NULL,

PERSON\_ID number(10) NOT NULL,

Job\_ID number(10) NOT NULL,

PRIMARY KEY(PEmployee\_ID),

FOREIGN KEY(Person\_ID) REFERENCES PERSON(Person\_ID),

FOREIGN KEY(PEmployee\_ID) REFERENCES INTERVIEW(Interviewee\_ID),

FOREIGN KEY (Job\_ID) REFERENCES JOB\_POSITION(Job\_ID)

);

CREATE TABLE JOB\_POSITION(

Job\_ID number(10) NOT NULL,

Descripton varchar(300) NOT NULL,

posted\_date date NOT NULL,

Dno number(10) NOT NULL,

PRIMARY KEY(Job\_ID),

FOREIGN KEY(Dno) REFERENCES DEPARTMENT(Dno)

);

CREATE TABLE POSTS\_JOB(

Job\_ID number(10) NOT NULL,

Dno number(10)NOT NULL,

PRIMARY KEY(Job\_ID,Dno),

FOREIGN KEY(Job\_ID) REFERENCES JOB\_POSITION(Job\_ID),

FOREIGN KEY(Dno) REFERENCES Department(Dno)

);

CREATE TABLE INTERVIEW(

Interview\_number number(10) NOT NULL,

Interviewer\_ID number(10) NOT NULL UNIQUE,

Interviewee\_ID number(10) NOT NULL UNIQUE,

Interview\_time number(4) NOT NULL,

Job\_position number(10) NOT NULL,

PRIMARY KEY(Interview\_number),

FOREIGN KEY(Interviewee\_ID) REFERENCES POTENTIAL\_EMPLOYEE(PEmployee\_ID),

FOREIGN KEY(Interviewer\_ID) REFERENCES EMPLOYEE(Employee\_ID),

FOREIGN KEY(Job\_position) REFERENCES JOB\_POSITION(Job\_ID)

);

CREATE TABLE ROUNDS(

Rounds number(1) NOT NULL,

Score number(3),

Interview\_number number(10) NOT NULL,

PRIMARY KEY(Rounds),

FOREIGN KEY(Interview\_number) REFERENCES INTERVIEW(Interview\_number)

);

CREATE TABLE PARTS(

Part\_ID number(10) NOT NULL,

price number(10) NOT NULL,

Product\_ID number(10) NOT NULL,

Vendor\_ID number(10) NOT NULL,

PRIMARY KEY(Part\_ID),

FOREIGN KEY(Product\_ID) REFERENCES PRODUCT(Product\_ID)

FOREIGN KEY(Vendor\_ID) REFERENCES VENDOR(vendor\_ID)

);

**D) CREATE VIEWS**

1)CREATE VIEW VIEW1 AS

SELECT E.employee\_ID, AVG(sp.amount) as average

FROM SALARY\_PAYMENT SP, employee E

GROUP BY sp.transaction\_number;

2) CREATE VIEW VIEW2 AS

SELECT i.interview\_number, i.interviewer, i.interviewee,COUNT(\*) as count

FROM ROUNDS r,INTERVIEW i

WHERE r.score > 60

GROUP BY i.interview number

3) CREATE VIEW VIEW3 AS

SELECT p.product\_ID, pt.type\_ID,COUNT(\*) as sales

FROM sale\_history sh,PRODUCT p, PRODUCT\_TYPE pt

GROUP BY sh.product\_ID;

4) CREATE VIEW VIEW4 AS

SELECT SUM(parts.price) as sum

FROM PARTS, PRODUCT p

GROUP BY p.type\_id;

**E) Queries**

**1)**

**SELECT i.interviewer\_ID,p.Fname,p.Lname**

**FROM Employee e**

**JOIN Person p**

**ON e.person\_id = p.person\_ID**

**JOIN interview i**

**ON e.employee\_ID = i.interviewer\_id**

**WHERE p.Fname= 'Hellen'AND p.Lname = 'Cole' AND i.job\_ID = 11111**

**;**

**2) select jp.job\_ID,jp.posted\_date**

**from job\_position jp**

**INNER JOIN department d**

**ON d.dno = jp.dno**

**WHERE d.dname= 'Marketing' AND jp.posted\_date > '01-jan-2011'**

**AND jp.posted\_date < '01-FEB-2011'**

**;**

**3) SELECT Employee\_ID, p.Fname, p.lname**

**FROM EMPLOYEE e**

**INNER JOIN PERSON p**

**ON p.person\_id = e.employee\_id**

**WHERE e.super\_ID <> e.employee\_id**

**;**

**4) SELECT ms.line1,ms.line2,ms.city,ms.state,ms.zipcode,COUNT(sh.site\_id) AS Sites**

**FROM sale\_history sh**

**JOIN Marketing\_site ms**

**ON sh.sale\_id = ms.sale\_id**

**WHERE sh.sale\_date > '01-May-2011' AND sh.sale\_date < '01-June-2011'**

**GROUP BY sh.site\_id**

**;**

**5)**

**SELECT jp.job\_ID, jp.job\_description**

**FROM job\_position jp**

**JOIN interview i**

**ON jp.job\_id = i.job\_id**

**JOIN rounds r**

**ON r.interview\_number = i.interview\_number**

**-- AVG(SCORE) as average < 60**

**WHERE AVG(score) < 60;**

**6)**

**SELECT sh.employee\_ID,p.Fnmae,p.Lname**

**FROM sales\_history sh, person p, product\_type pt**

**WHERE Sh.employee\_ID = p.person\_ID AND pt.list\_price >= 200**

**GROUP BY sh.employee\_ID,p.Fnmae,p.Lname**

**HAVING COUNT(Unique sh.product\_ID) >=**

**(SELECT COUNT(pt.product\_ID)**

**FROM PRODUCT\_TYPE pt WHERE pt.list\_price >=200);**

**SELECT D.Dno, d.dname**

**FROM DEPARTMENT d**

**WHERE d.Dno NOT in (SELECT job\_ID from Job\_position jp**

**WHERE Jp.posted\_date > '01 - jan -2011' AND jp.posted\_date < '31-jan-2011');**

**SELECT p.person\_ID,p.Fname,p.Lname, e.Dno**

**FROM PERSON p, Employee e, POTENTIAL\_EMPLOYEE pe**

**WHERE p.person\_ID = e.employee\_ID AND p.person\_ID = pe.PEmployee\_ID AND pe.job\_id =12345;**

**12) SELECT p.Email, P.Fname, P.Lname, AVG(I.grade) as average**

**FROM PERSON p, INTERVIEW I, Rounds r**

**WHERE p.person\_ID = I.interviewee\_ID AND r.interview\_number = i.interview\_number and r.Round >=5**

**GROUP BY I.interviewee\_ID, I.job\_ID, P.Fname, P.Lname, P.Email**

**HAVING AVG(r.score) >= 70;**

**13) SELECT I.interviewee\_ID,p.Fname, p.Lname,p.email,p.phnum1**

**FROM INTERVIEW i, PERSON p, Rounds r**

**WHERE i.interviewee\_ID = p.person\_ID and r.interview\_number = i.interview\_number and**

**r.round >=5**

**GROUP BY i.intervee\_ID,p.Fname, p.Lname,p.email,p.phnum1**

**HAVING AVG(r.score) >= 70 and COUNT(unique i.job\_ID) = (SELECT COUNT(PE.pemployee\_ID)**

**from Potential\_Employee PE WHERE PE.pEmployee\_ID= i.interviewee\_ID);**

**15) Select v.vendor\_ID, v.Vname**

**FROM vendor v, part p**

**WHERE v.vendor\_ID = p.vendor\_ID AND p.part\_type = 'cup' and p.weight < 4**

**ORDER BY p.part\_price ASC**

**FETCH FIRST 1 ROW with ties;**

**DEPENDENCY DIAGRAM – (P.T.O)**

