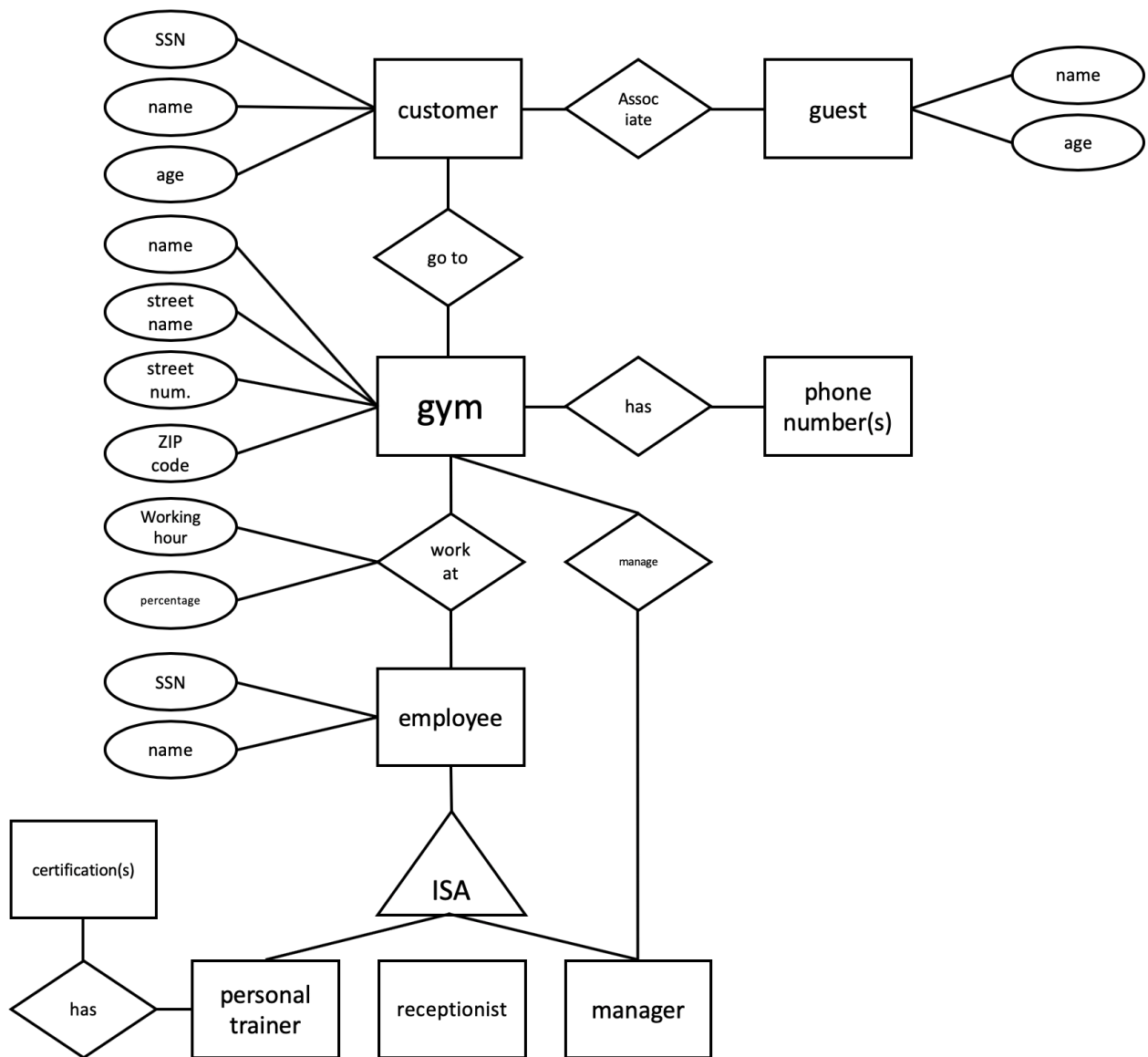


Web Application Assignment 1

Ruiyu Zhang || rz213
2019.02.05

Question 1

1) ER-Diagram



2) SQL codes that creates database

```
CREATE TABLE gym (  
    gym_id BIGINT NOT NULL AUTO_INCREMENT,  
    name CHAR(30) NOT NULL,  
    street_name CHAR(30),  
    street_num CHAR(10),  
    zip_code CHAR(5),  
    manager CHAR(11) NOT NULL,  
    FOREIGN KEY (manager) REFERENCES employee (SSN),  
    PRIMARY KEY (gym_id)  
)
```

```
CREATE TABLE employee (  
    emp_id BIGINT NOT NULL AUTO_INCREMENT,  
    ssn CHAR(11) NOT NULL,  
    name CHAR(30),  
    specialization CHAR(20),  
    PRIMARY KEY (emp_id)  
)
```

```
CREATE TABLE customer (  
    uid BIGINT NOT NULL AUTO_INCREMENT,  
    ssn CHAR(11) NOT NULL,  
    name CHAR(30),  
    age INT,  
    PRIMARY KEY (uid)  
)
```

```
CREATE TABLE guest (  
    uid BIGINT,  
    age INT,  
    name CHAR(30),  
    FOREIGN KEY (customer) REFERENCES (uid)  
)
```

```
CREATE TABLE phone (  
    pho_id BIGINT NOT NULL AUTO_INCREMENT,  
    phone_num BIGINT NOT NULL,  
    gym_id BIGINT,  
    FOREIGN KEY (gym_id) REFERENCES gym (gym_id)  
)
```

```
CREATE TABLE go_to (  
    uid BIGINT,  
    gym_id BIGINT,  
    PRIMARY KEY (uid, gym_id),  
    FOREIGN KEY (gym_id) REFERENCES gym (gym_id),
```

```

        FOREIGN KEY (uid) REFERENCES customer (uid)
    )

    CREATE TABLE work_at (
        gym_id BIGINT NOT NULL,
        emp_id BIGINT NOT NULL,
        percentage REAL,
        working_hours CHAR(20),
        PRIMARY KEY (gym_id, emp_id),
        FOREIGN KEY (gym_id) REFERENCES gym (gym_id),
        FOREIGN KEY (emp_id) REFERENCES employ (emp_id)
    )

    CREATE TABLE certification (
        cer_id BIGINT NOT NULL AUTO INCREMENT,
        emp_id BIGINT NOT NULL,
        certification_title CHAR(30),
        FOREIGN KEY (emp_id) REFERENCES empolyee (emp_id)
    )

```

Question 2

1)

```

SELECT s.sname FROM Suppliers s
WHERE NOT EXISTS ( -- true if 'all parts' included in 'all parts from supplier'
    SELECT p.pid FROM Parts p EXCEPT ( -- all kinds of parts
        SELECT c.pid FROM Catalog c -- all kinds of parts from current supplier
        WHERE c.sid = s.sid
    )
)

```

2)

```

SELECT DISTINCT c1.sid FROM Catalog c1
WHERE c1.cost > (
    SELECT AVG(c2.cost) FROM Catalog c2 -- average of all of current kind of
part
    WHERE c1.pid = c2.pid
)

```

3)

```
SELECT s.sname FROM Suppliers s, Catalog c1
WHERE c1.sid = s.sid AND c1.cost = (
    SELECT MAX(c2.cost) FROM Catalog c2 -- max of all of current kind of part
    WHERE c1.pid = c2.pid
)
```

4)

```
SELECT c.sid FROM Catalog c
WHERE NOT EXISTS (
    SELECT p.color FROM Parts p -- all colors (except red) of parts from cur
    supplier
    WHERE p.pid = c.pid AND p.color <> "red"
)
```

5)

```
SELECT c.sid FROM Catalog c
WHERE EXISTS (
    SELECT p.color FROM Parts p
    WHERE p.pid = c.pid AND (p.color = "red" AND p.color = "green")
)
```

6)

```
SELECT s.sname, MAX(c.cost)
FROM Suppliers s, Catalog c, Parts p
WHERE c.sid = s.sid, c.pid = p.pid
    AND p.color IN ("red", "green")
```

Question 3

1)

```

SELECT m.MovieName
FROM Movies m, MovieSupplier ms, Suppliers s
WHERE ms.SupplierID = s.SupplierID AND ms.MovieID = m.MovieID
      AND (s.SupplierName = "Ben's Video" OR s.SupplierName = "Video Clubhouse")

```

2)

```

SELECT m.MovieName
FROM Movies m, Inventory i, Rentals r
WHERE i.MovieID = m.MovieID AND i.TapeID = r.TapeID
      AND r.Duration >= ALL(
          SELECT Duration FROM Rentals
      )

```

3)

```

SELECT s.SupplierName
FROM Supplier s
WHERE s.SupplierID NOT IN (
    SELECT ms.SupplierID
    FROM MovieSupplier ms, Inventory i
    WHERE NOT EXISTS (
        SELECT *
        FROM MovieSupplier ms2, Inventory i2
        WHERE i2.MovieID = ms2.MovieID
              AND i2.MovieID = i.movieID
              AND ms2.MovieID = ms.MovieID
    )
)

```