System Metropolitan State University
ICS 311-50 Summer 2023 Database Management System
HWPS1

Due Date: 6/12/2023
Total Points: 40

Question 1 (4 Points)

Given the table structure shown below, answer the following questions:

team_code	team_lead_name	team_lead_phone	team_lead_address	team_income	team_lead_pay	hire_date
21-5Z	Alley C. Smith	904-338-3416	3334 Lee Rd., Gainsville, FL 37123	16833460.00	75000.00	01062015
25-2D	Jane D. Grant	615-898-9909	218 Clark Blvd., Nashville, TN 36362	12500000.00	78000.00	03012010
25-5A	George F. Dorts	615-227-1245	124 River Dr., Franklin, TN 29185	32512420.00	77000.00	12252012
25-9T	Alley C. Smith	904-338-3416	3334 Lee Rd., Gainsville, FL 37123	21563234.00	75000.00	01062015
27-4Q	George F. Dorts	615-227-1245	124 River Dr., Franklin, TN 29185	10314545.00	77000.00	12252012
29-2D	Alley C. Smith	904-338-3416	3334 Lee Rd., Gainsville, FL 37123	25559999.00	75000.00	01062015
31-7P	William K. Moor	904-445-2719	216 Morton Rd., Stetson, FL 30155	56850000.00	79000.00	11212014

1. How many tuples does the table contain? How many attributes are there per tuple?
   - 7 tuples, with 7 attributes

2. What data redundancies can you detect in the table? Explain why data redundancy is undesired?

- the team lead name of Alley C. Smith is entered 3 times, of George Doris 2 times, with the same phone number, address, salary, and hired date, however, with different team_cde data
- Its undesired because it requires time and memory, and it makes it more confusing

Question 2 (6 points)

Describe (use good table layout format/style like those used in the lecture slides, i.e. Word table format) at least 3 tables (with some attributes) that might be used to store information in a Car Dealership system.

Buyer	Seller	Brand	
date_sold	seller_name	car_brand	
Price	seller_id	car_color	
buyer_name	cars_sold	car_date	
buyer_address	total_price	car_miles	

Question 3 (10 points)

Consider the database schema below:
Notes: We use a simplified schema where we assume customer names are unique. Just in this schema.

branch(branch_name, branch city, assets)
customer (customer_name, customer street, customer city)
loan (loan_number, branch_name, amount)
borrower (customer_name, loan_number)
account (account_number, branch_name, balance)
depositor (customer_name, account_number)

a) What are the appropriate primary keys? (6 points)

- loan_number

Given your choice of primary keys, identify appropriate foreign keys. (4 points)

- account_number, customer_name, branch_name, balance

Question 4 (6 points)

Using the database above, give an expression in the relational algebra for each of the following queries:

a) Find all loan numbers with a loan value greater than $10,000.

- select loan_number
- From loan
- Where amount > 10000;

b) Find the names of all depositors who have an account with a value greater than $6,000.

- select customer_name
- From depositor d

```
- Join account a
- On d.account_number = a.account_number
- Where balance > 6000;
```

c) Find the names of all depositors who have an account with a value greater than $6,000 at the "Uptown" branch.

```
- select customer_name
- From depositor d
- Join account a
- On d.account_number = a.account_number
- Where balance > 6000 AND branch_name like '%Uptown%';
```

Question 5 (10 points)

For the database schema below, write SQL DDL corresponding to the schema. Make any reasonable assumptions about data types, and be sure to declare primary and foreign keys.

person (driver id, name, address)
car (license, model, year)
accident (report number, date, location)
owns (driver id, license)
participated (report number, license, driver id, damage amount)

```
CREATE TABLE person(
        driver_id int not null primary key;
        name varchar(50),
        address varchar(100)
);

CREATE TABLE car(
        license varchar(50),
        model varchar(50),
        year int
);

CREATE TABLE accident(
        report_number int,
        date DATE,
        location varchar(50)
);

CREATE TABLE owns(
        driver_id int,
        license varchar(50),
        PRIMARY KEY(driver_id, license),
        FOREIGN KEY(driver_id) REFERENCES person(driver_id),
```

```
        FOREIGN KEY(license) REFERENCES car(license)
);

CREATE TABLE participated(
        report_number int not null,
        license varchar(50),
        driver_id not null,
        damage_amount int,
        PRIMARY KEY(license, driver_id),
        FOREIGN KEY(report_number) REFERENCES accident(report_number)
);
```

Question 6 (4 points)

Using the university schema in our textbook, write the following queries in SQ:

a) Create a new course “CS-001”, titled “Weekly Seminar”, with 0 credits.

```
- insert into course(course_id, title, dept_name, credits) values ('CS-001', 'Weekly Seminar', 'Comp. Sci.', 0);
```

b) Create a section of this course in Autumn 2009, with *section id* of 1.

```
-insert into section(course_id, sec_id, semester, year) values ('CS-001', '1', 'Autumn', 2009);
```