SQL Review

Quiz Format

- **Closed book**
- **Part1: Multiple choice & short answer questions**
  - 30 minutes, 20 questions * 3 points each = 60 points
- **Part2: SQL queries**
  - 60 minutes, 10 questions * 4 points each = 40 points
  - 10 `SELECT` queries
  - Based on a provided database schema and sample data

Quiz Prep Tips

1. **Review the slides and code examples (No DataCamp unless covered in class)**
2. **Mock exam on Ed**
3. **Post questions on Ed**

Basic SELECT Syntax

SELECT all columns and rows from person table

```
SELECT * FROM person;
```

Limit number of rows returned

```
SELECT * FROM person LIMIT 5;
```

Order rows by a specific column

```
SELECT * FROM person ORDER BY name;
SELECT * FROM person ORDER BY id DESC;
```

Select specific columns from person table

```
SELECT id, name FROM person;
```

Basic SELECT Syntax

Select specific rows from person table

```
SELECT * FROM person WHERE id = 1;
SELECT * FROM person WHERE age > 30;
SELECT * FROM person WHERE age BETWEEN 20 AND 30;
```

Select rows with pattern matching

```
SELECT * FROM person WHERE name LIKE 'Alice%';
SELECT * FROM person WHERE id LIKE '_H42W%';
```

Advanced SELECT Syntax

Aggregate functions

```
SELECT COUNT(*) FROM person;
SELECT MIN(age) FROM person;
SELECT MAX(age) FROM person;
```

Aggregate functions and GROUP BY

```
SELECT person_id, COUNT(*) AS checkin_count
FROM facebook_event_checkin GROUP BY person_id;
```

Having clause

```
SELECT person_id
FROM facebook_event_checkin GROUP BY person_id
HAVING COUNT(*) > 1;
```

Advanced SELECT Syntax

Subquery to reuse query results

```
SELECT id, name, ssn
FROM person
WHERE license_id IN (
    SELECT id FROM driver_license WHERE hair_color = 'red');
```

JOINs to combine tables

```
-- without table aliases
SELECT person.id, person.name, drivers_license.hair_color
FROM person JOIN drivers_license ON person.license_id = drivers_license.id
WHERE drivers_license.hair_color = 'red';
-- with table aliases
SELECT p.id, p.name, dl.hair_color
FROM person AS p JOIN drivers_license AS dl ON p.license_id = dl.id
WHERE dl.hair_color = 'red';
```

DDL & DML to Create and Modify Tables

```
DROP TABLE IF EXISTS movies;
CREATE TABLE movies (
    id INT,
    main_char_id INT,
    title text
);

ALTER TABLE movies ADD COLUMN year INT;
ALTER TABLE movies DROP COLUMN year;

ALTER TABLE movies ADD CONSTRAINT pk_movies PRIMARY KEY (id);

ALTER TABLE movies
ADD CONSTRAINT fk_main_char FOREIGN KEY (main_char_id) REFERENCES students(id);
```

DDL & DML to Create and Modify Tables

```sql
-- continued from previous slide
INSERT INTO movies (id, main_char_id, title)
VALUES (1, 101, 'The Great Adventure'),
       (2, 102, 'Another Story');

UPDATE movies SET title = 'The Greatest Adventure' WHERE id = 1;

DELETE FROM movies WHERE id = 1;

-- FK constraint violation if 103 not in students.id
INSERT INTO movies VALUES (3, 103, 'New Movie');

DROP TABLE movies;
```