Group Functions

- Get the average patient’s age

  ```
  SELECT AVG(DATEDIFF(NOW(), bdate)/365) FROM Patient;
  ```

- Can you find the birthdates of the oldest and youngest patients?

  ```
  SELECT MIN(bdate), MAX(bdate) FROM Patient;
  ```

- Can you find the first and last appointment time?

  ```
  SELECT
    MIN(TIME(a_date)) 'First Appt’,
    MAX(TIME(a_date)) 'Last Appt’
  FROM Appointment;
  ```

Grouping Rows – Using GROUP BY

- How many total appointments do Dr. Adams and Dr. Cheatle have at 10am and 11am?

```
SELECT d.d_lastname, TIME(a.a_date), COUNT(*) FROM Doctor d
JOIN Appointment a
WHERE (d.d_lastname = 'Adams' OR d_lastname = 'Cheatle’)
AND (TIME(a.a_date) = '10:00:00' OR TIME(a.a_date) = '11:00:00’)
GROUP BY d.d_lastname, TIME(a.a_date);
```

- What is the average patient age per specialty?

```
SELECT d.specialty,
  ROUND(AVG(DATEDIFF(NOW(), p.bdate)/365), 2)
FROM Patient p
JOIN Appointment a ON p.p_id = a.p_id
JOIN Doctor d ON a.d_id = d.d_id
GROUP BY d.specialty;
```

Restricting Aggregated Output – Using HAVING

- Syntax:
```
SELECT column1, AGGREGATE_FUNCTION(column2) FROM table
GROUP BY column1
HAVING AGGREGATE_FUNCTION(column2) condition;
```
    - *condition* is a condition to be satisfied by the aggregated value
- Which doctors have more than 2 appointments?
```
SELECT d.d_id, COUNT(*) AS 'Num of Appts’ FROM Doctor d
JOIN Appointment a ON d.d_id = a.d_id
GROUP BY d.d_id
HAVING COUNT(*) > 2;
```
- Which specialties have more than 1 doctor?
```
SELECT specialty, count(d_id) UniqueDocs FROM Doctor
GROUP BY specialty
HAVING UniqueDocs > 1;
```

Restricting Aggregated Output – Using HAVING

- Which specialties have an average patient age of greater than 32 years old?

```
SELECT d.specialty, ROUND(AVG(DATEDIFF(NOW(), p.bdate)/365),
2) FROM Patient p
JOIN Appointment a ON p.p_id = a.p_id
JOIN Doctor d ON a.d_id = d.d_id
GROUP BY d.specialty
HAVING AVG(DATEDIFF(NOW(), p.bdate)/365) > 32;
```