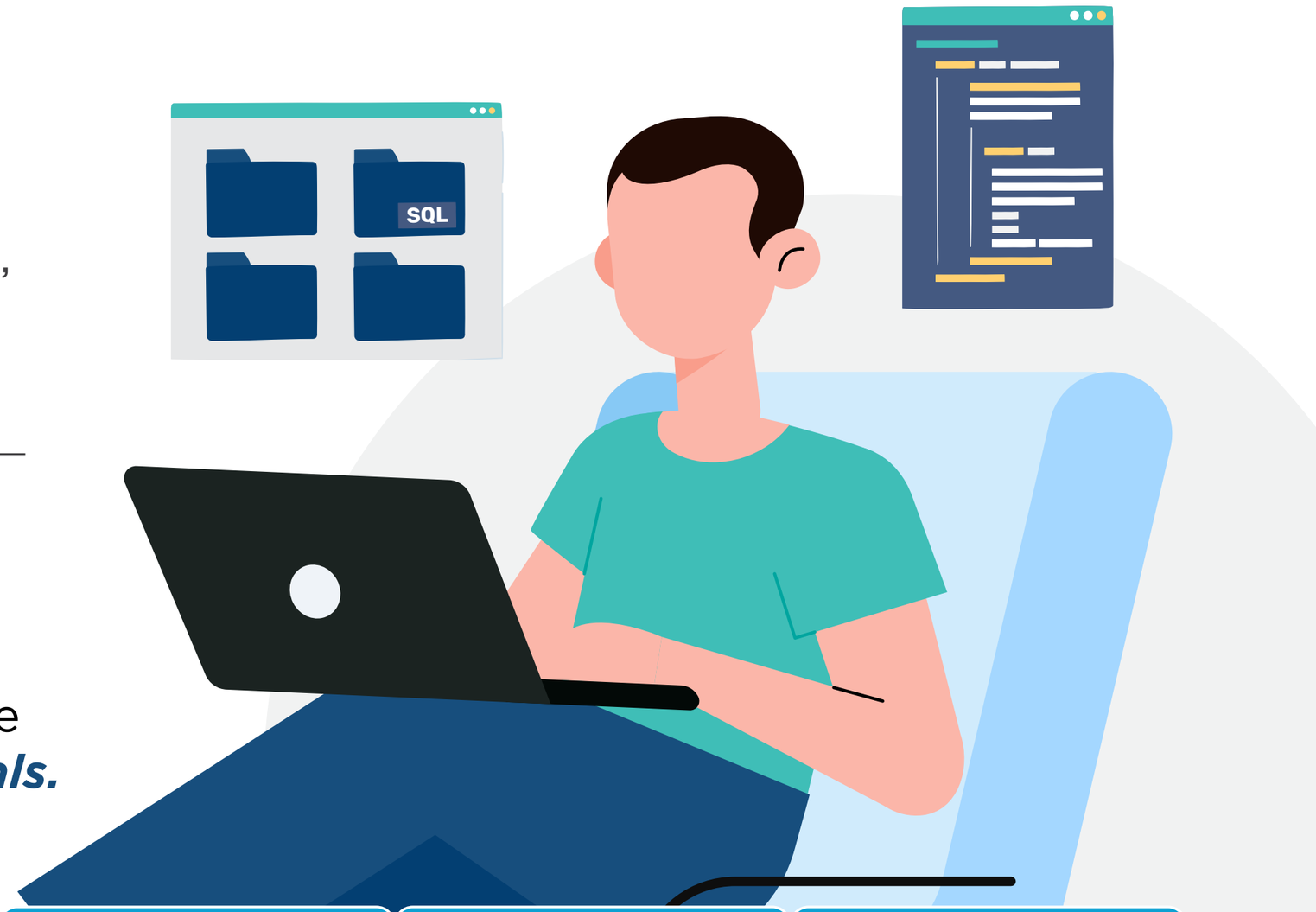
SQL Cheat Sheet

SQL stands for Structured Query Language, it allows you to access and manipulate databases and is the standard language for storing, manipulating, and retrieving data in databases.



Database Table

Throughout this cheat sheet, we'll reference the columns listed in the sample table of ***cornell_car_rentals.***

owner.id	location.city	location.state	RenterTrips Taken	fuelType
12847615	Seattle	WA	13	ELECTRIC
10199256	Albuquerque	NM	28	HYBRID
2842845	Beech Island	SC	21	GASOLINE
6025007	Atlanta	GA	52	DIESEL

The table above contains four records (one for each customer) and five columns (owner.id, location.city, location.state, renterTripsTaken, and fuelType).

A. Querying Tables (SELECT Statement)

1. The SELECT statement is used to select data from a database.

a

SELECT all columns from the table:
returns all columns from cornell_car_rentals

```
SELECT *
FROM cornell_car_rentals;
```

b

SELECT a single column from the table:
returns city location (location.city) from cornell_car_rentals

```
SELECT *
FROM cornell_car_rentals;
```

c

SELECT multiple columns from the table:
returns city location and trips taken (renterTripsTaken)

```
SELECT location.city, renterTripsTaken
FROM cornell_car_rentals;
```

d

SELECT DISTINCT returns only distinct (different) values: returns a unique list of city locations from the table

```
SELECT DISTINCT location.city
FROM cornell_car_rentals;
```

e

The ORDER BY keyword is used to sort results either in ascending or descending order ASC/DESC : selects **owner.id , location . city** , ordered by the **renterTripsTaken** in ascending order

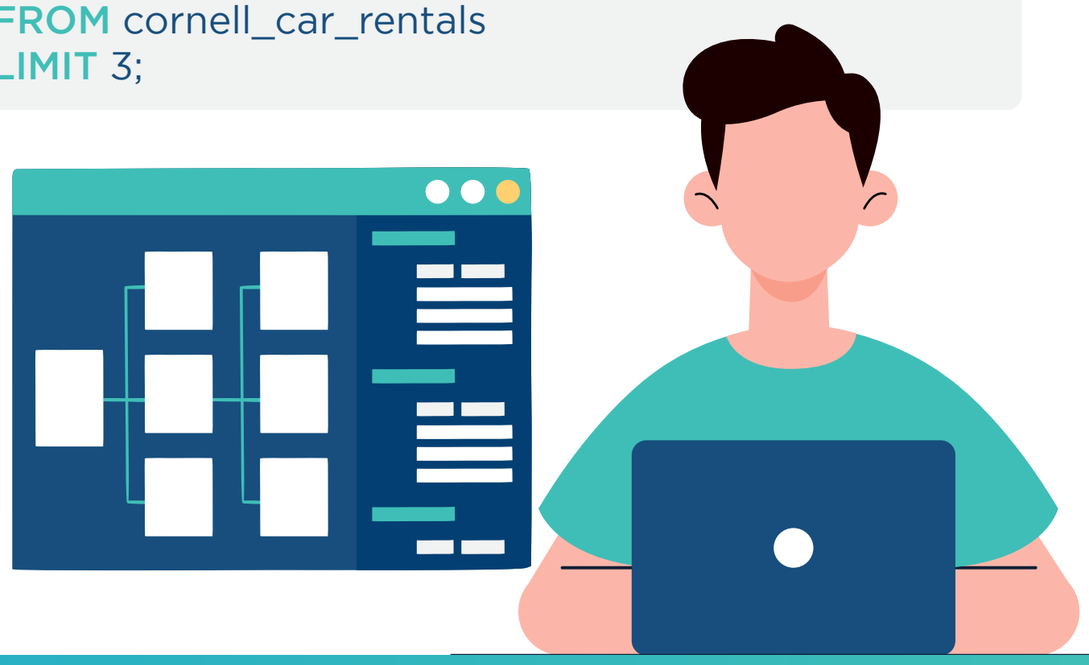
```
SELECT owner.id location.city
FROM cornell_car_rentals
ORDER BY renterTripsTaken ASC;
```

```
SELECT owner.id location.city
FROM cornell_car_rentals
ORDER BY renterTripsTaken DESC;
```

f

The LIMIT keyword is used to return a specific number of rows: returns the top 3 rows from the table.

```
SELECT *
FROM cornell_car_rentals
LIMIT 3;
```



B. Filter Operation Data (WHERE)

The WHERE statement is used to filter records. It is used to extract only those records that meet a specified condition.

```
SELECT *
FROM cornell_car_rentals
WHERE fuelType='HYBRID';
```

```
SELECT *
FROM cornell_car_rentals
WHERE renterTripsTaken=13;
```

```
SELECT *
FROM cornell_car_rentals
WHERE renterTripsTaken > 30;
```

```
SELECT *
FROM cornell_car_rentals
WHERE renterTripsTaken >= 21;
```

```
SELECT *
FROM cornell_car_rentals
WHERE renterTripsTaken BETWEEN 13 AND 50;
```

```
SELECT *
FROM cornell_car_rentals
WHERE location.state IN ('WA', 'SC');
```

The WHERE clause can be combined with logic operators (AND , OR , and NOT)to filter records based on the condition.

a

AND selects all fields from **"cornell_car_rentals"** where fuel type is **"HYBRID"** AND city is **"Albuquerque"**:

```
SELECT *
FROM cornell_car_rentals
WHERE fuelType="HYBRID" AND
location.city="Albuquerque";
```

b

OR selects all fields from **"cornell_car_rentals"** where city is **"Seattle"** OR **"Atlanta"** :

```
SELECT *
FROM cornell_car_rentals
WHERE location.city="Seattle" OR
location.city="Atlanta";
```

c

NOT selects all fields from **"cornell_car_rentals"** where city is NOT **"Beech Island"**:

```
SELECT *
FROM cornell_car_rentals
WHERE NOT Country="Beech Island";
```

d

Combining Operators: Get all the listings where the city starts with and where the cit 'a' and does not end in 'e'

```
SELECT *
FROM cornell_car_rentals
WHERE location.city LIKE 'a%' AND NOT LIKE '%e';
```

C. SQL Aliases (AS)

SQL aliases are used to give a table, or a column in a table, a temporary name. An alias is created using the AS keyword.

a

Alias Column Syntax

```
SELECT owner.id AS ID, location.city AS City
FROM cornell_car_rentals;
```

b

Alias Table Syntax

```
SELECT owner.id
FROM cornell_car_rentals AS rentals;
```

D. SQL Join (AS)

- **INNER JOIN :** Returns records that have matching values in both tables
- **LEFT JOIN :** Returns all records from the left table matching the records from the right table
- **RIGHT JOIN :** Returns all records from the right table matching the recordsfrom the left table
- **FULL JOIN :** Returns all records when there is a match in either left or right table

INNER JOIN syntax

```
SELECT column_name(s)
FROM table1
INNER JOIN table2
ON table1.column_name = table2.column_name;
```

RIGHT JOIN syntax

```
SELECT column_name(s)
FROM table1
RIGHT JOIN table2
ON table1.column_name = table2.column_name;
```

LEFT JOIN syntax

```
SELECT column_name(s)
FROM table1
LEFT JOIN table2
ON table1.column_name = table2.column_name;
```

FULL JOIN syntax

```
SELECT column_name(s)
FROM table1
FULL JOIN table2
ON table1.column_name = table2.column_name;
```