codecademy / PRO

INTENSIVE

Learn SQL from Scratch

Aggregate Functions

Calculations performed on multiple rows

```
SELECT COUNT(*)
FROM fake_apps;
```

COUNT()

Calculate how many rows are in the `fake_apps` table

```
SELECT SUM(downloads)
FROM fake_apps;
```

SUM()

Calculate the sum of all the values in the `downloads` column

```
SELECT MAX(downloads)
FROM fake_apps;
```

MAX()

Return the highest value in the `downloads` column

```
SELECT MIN(downloads)
FROM fake_apps;
```

MIN()

Return the lowest value in the `downloads` column

```
SELECT AVG(price)
FROM fake_apps;
```

AVG()

Calculate the average value of the `price` column

```
SELECT category, SUM (downloads)
FROM fake_apps
GROUP BY category;
```

GROUP BY

Calculate the total number of `downloads` for each `category`

```
SELECT category, SUM(downloads)
FROM fake_apps
GROUP BY 1;
```

GROUP BY

The reference number `1` refers to the first selected column (`category`)

```
SELECT category, SUM(downloads)
FROM fake_apps
GROUP BY category
HAVING SUM(downloads) > 5;
```

HAVING

`WHERE` can not be used with aggregate functions

`HAVING` is used to filter groups

```sql
SELECT category, SUM(downloads)
FROM fake_apps
WHERE price = 0
GROUP BY 1
HAVING SUM(downloads) > 2
ORDER BY 2 DESC
LIMIT 5;
```

Review

1. `SELECT` as many columns as you want
2. `FROM` indicates which table
3. `WHERE` filters based on individual rows
4. `GROUP BY` tells you how to bucket your data
5. `HAVING` filters based on your aggregates
6. `ORDER BY` sorts the query
7. `LIMIT` reduces the number of results