

The Sequel to SQL

Level 1 – Section 2

Aggregate Functions Within SQL Clauses

Using Aggregate Functions Within SQL Clauses

How can we find the cost of all the Romance and Horror movies, grouped by genre?

Movies		
title	cost	genre
Gone With the Wind	390,000	Romance
Casablanca	1,039,000	Romance
Frankenstein	3,000,000	Horror
Creature From the Black Lagoon	500,000	Horror

```
SELECT sum(cost)
FROM Movies;
```



sum

4929000.00
(1 row)



Not what we want.

Using the GROUP BY Clause

GROUP BY will group together the cost for all movies with similar genres.

Movies

title	cost	genre
Gone With the Wind	390,000	Romance
Casablanca	1,039,000	Romance
Frankenstein	3,000,000	Horror
Creature From the Black Lagoon	500,000	Horror

```
SELECT genre, sum(cost)
FROM Movies
GROUP BY genre;
```



genre	sum
Horror	3500000.00
Romance	1429000.00
(2 rows)	



Introducing the GROUP BY Clause

We can use the GROUP BY clause to condense a group of columns into a single row.

GROUP BY Recipe

```
SELECT column_name, aggregate_function(column_name)
FROM table_name
GROUP BY column_name;
```


Filtering Aggregate Functions

Movies

title	cost	genre
Gone With the Wind	390,000	Romance
Casablanca	1,039,000	Romance
Casanegra	1,000,000	Romance
Frankenstein	3,000,000	Horror
Creature From the Black Lagoon	500,000	Horror
Not So Funny Movie	100	Comedy
Some Scary Movie	200,000	Horror

```
SELECT genre, sum(cost)
FROM Movies
GROUP BY genre;
```

genre	sum
Horror	3700000.00
Romance	2429000.00
Comedy (3 rows)	100.00

Only **1** Comedy film...

What if we only wanted genres that appear more than once to be part of our result?

Only groups that had 2 genres in it.

Using the HAVING Clause

We can use the HAVING clause to only include genres that have more than 1 movie.

Movies

title	cost	genre
Gone With the Wind	390,000	Romance
Casablanca	1,039,000	Romance
Casanegra	1,000,000	Romance
Frankenstein	3,000,000	Horror
Creature from the Black Lagoon	500,000	Horror
Not So Funny Movie	100	Comedy
Some Scary Movie	200,000	Horror

More than 1 row per group

```
SELECT genre, sum(cost)
FROM Movies
GROUP BY genre
HAVING COUNT(*) > 1;
```



genre	sum
Horror	3700000.00
Romance	2429000.00
(2 rows)	

Introducing the HAVING Clause

The HAVING clause restricts the groups of rows to only those who meet the specified condition.

HAVING Recipe

```
SELECT column_name, aggregate_function(column_name)
FROM table_name
WHERE column_name operator value (optional)
GROUP BY column_name
HAVING aggregate_function (column_name) operator value;
```



The **HAVING** clause

Using WHERE With the HAVING Clause

We can use WHERE to filter individual rows before the application of GROUP BY.

Movies

title	cost	genre
Gone With the Wind	390,000	Romance
Casablanca	1,039,000	Romance
Casanegra	1,000,000	Romance
Frankenstein	3,000,000	Horror
Creature from the Black Lagoon	1,000,000	Horror
Not So Funny Movie	2,000,000	Comedy
Some Scary Movie	200,000	Horror

If we only wanted to count movies that grossed at least \$1 million:

```
SELECT genre, sum(cost)
FROM Movies
WHERE cost >= 1000000
GROUP BY genre
HAVING COUNT(*) > 1;
```

genre	sum
Romance	2039000
Horror	4000000
(2 rows)	

