

## ORDER BY

Ans==> Alphabetically (A-Z)

### Sorting single columns

1. Get the names of people from the people table, sorted alphabetically.

```
SELECT name
```

```
FROM people
```

```
ORDER BY name ASC;
```

2. Get the names of people, sorted by birth date.

```
SELECT name
```

```
FROM people
```

```
ORDER BY birthdate ASC;
```

3. Get the birth date and name for every person, in order of when they were born.

```
SELECT name, birthdate
```

```
FROM people
```

```
ORDER BY birthdate ASC;
```

### Sorting single columns (2)

1. Get the title of films released in 2000 or 2012, in the order they were released.

```
SELECT title
```

```
FROM films
```

```
where release_year=2000 or release_year=2012
```

```
ORDER BY release_year ASC;
```

2. Get all details for all films except those released in 2015 and order them by duration.

```
SELECT *
```

```
FROM films
```

```
where release_year <> 2015
```

ORDER BY duration ASC;

3. Get the title and gross earnings for movies which begin with the letter 'M' and order the results alphabetically.

SELECT title, gross

FROM films

where title like 'M%'

ORDER BY title ASC;

## Sorting single columns (DESC)

1. Get the IMDB score and film ID for every film from the reviews table, sorted from highest to lowest score.

select imdb\_score, film\_id

from reviews

order by imdb\_score DESC;

2. Get the title for every film, in reverse order.

select title

from films

order by title DESC;

3. Get the title and duration for every film, in order of longest duration to shortest.

select title, duration

from films

order by duration DESC;

## Sorting multiple columns

1. Get the birth date and name of people in the people table, in order of when they were born and alphabetically by name.

select birthdate, name

from people

order by birthdate, name;

2. Get the release year, duration, and title of films ordered by their release year and duration.

```
select release_year, duration, title
```

```
from films
```

```
order by release_year, duration;
```

3. Get certifications, release years, and titles of films ordered by certification (alphabetically) and release year.

```
select certification, release_year, title
```

```
from films
```

```
order by certification ASC, release_year;
```

4. Get the names and birthdates of people ordered by name and birth date.

```
select name, birthdate
```

```
from people
```

```
order by name, birthdate;
```

## GROUP BY

Ans ==> Performing operations by group

## GROUP BY practice

1. Get the release year and count of films released in each year.

```
SELECT release_year, count(*)
```

```
FROM films
```

```
GROUP BY release_year;
```

2. Get the release year and average duration of all films, grouped by release year.

```
SELECT release_year, avg(duration)
```

```
FROM films
```

```
GROUP BY release_year;
```

3. Get the release year and largest budget for all films, grouped by release year.

```
SELECT release_year, max(budget)
```

```
FROM films
```

```
GROUP BY release_year;
```

4. Get the IMDB score and count of film reviews grouped by IMDB score in the reviews table.

```
SELECT imdb_score, count(*)  
FROM reviews  
GROUP BY imdb_score;
```

## GROUP BY practice (2)

1. Get the release year and lowest gross earnings per release year.

```
select release_year, min(gross)  
from films  
group by release_year
```

2. Get the language and total gross amount films in each language made.

```
select language, sum(gross)  
from films  
group by language
```

3. Get the country and total budget spent making movies in each country.

```
select country, sum(budget)  
from films  
group by country
```

4. Get the release year, country, and highest budget spent making a film for each year, for each country. Sort your results by release year and country.

```
select release_year, country, max(budget)  
from films  
group by release_year, country  
order by release_year, country
```

5. Get the country, release year, and lowest amount grossed per release year per country. Order your results by country and release year.

```
select release_year, country, min(gross)  
from films  
group by release_year, country
```

order by country, release\_year;

## HAVING a great time

Ans ==> 13

## All together now

1. Get the release year, budget and gross earnings for each film in the `films` table.

```
select release_year, budget, gross
from films;
```

## All together now (2)

1. Get the country, average budget, and average gross take of countries that have made more than 10 films. Order the result by country name, and limit the number of results displayed to 5. You should alias the averages as `avg_budget` and `avg_gross` respectively.

```
-- select country, average budget, average gross
select country, avg(budget) as avg_budget, avg(gross) as avg_gross
-- from the films table
from films
-- group by country
group by country
-- where the country has more than 10 titles
having count(title) > 10
-- order by country
order by country
-- limit to only show 5 results
limit 5
```

## A taste of things to come

Submit the code in the editor and inspect the results.

```
SELECT title, imdb_score
FROM films
JOIN reviews
ON films.id = reviews.film_id
WHERE title = 'To Kill a Mockingbird';
```