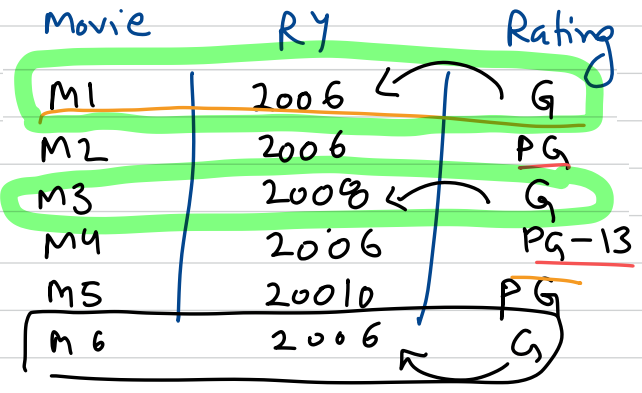



G - 2008

G - 2006

PG - 2006

PG - 2010

PG-13 - 2006

all Cols

```
Select Distinct rating, year, movieName
FROM film;
```

**What does the DISTINCT keyword do in a SELECT statement?**

38 users have participated

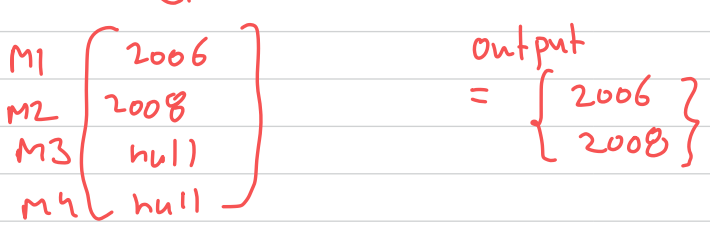
| | | |
|---|---|---|
| A | It counts the number of unique records in a column. | 37% ← X |
| B | It finds the sum of all records in a column. | 0% |
| ✓ C | It eliminates duplicate records in the output. | 61% ← ✓ |
| D | It sorts the records in ascending order. | 3% |

End Quiz Now

**What is the result of the following SQL query: `SELECT DISTINCT column1 FROM table1;` ?**

43 users have participated

| | | |
|---|---|---|
| A | It displays all values of column1, including duplicates. | 7% |
| B | It displays unique non-null values of column1. | 86% |
| C | It counts the total number of unique values in column1. | 7% |
| D | It sorts all values in column1. | 0% |



set { 1, 2, 3, 1, 4, 1 }

↓

(1, 2, 3, 4)

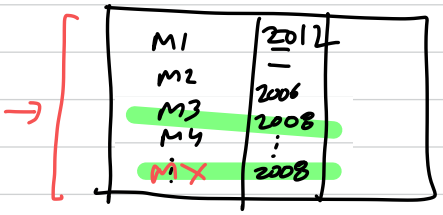
set ( (M1, null)
→ (M2, 2008)
(M3, 2008)
(M3, 2008) )

↓

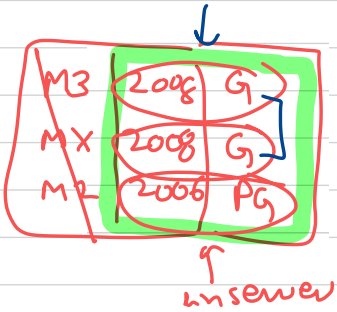
(M1, null)
(M2, 2008)
(M3, 2008)

Select DISTINCT year, movie

# WHERE CLAUSE




- answer = []
- for row in film:
  if (year == 2008 OR year = 2006)
  answer.append (row[rating], row[releyr])

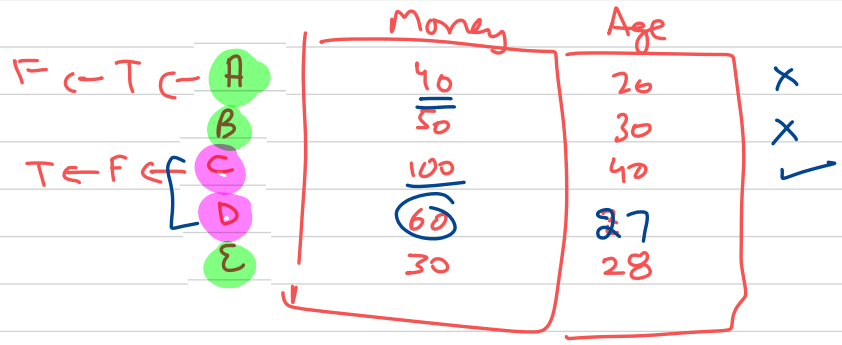



```
SELECT DISTINCT rating, releaseYear
FROM film
WHERE releaseyear = 2008,
```

WHERE
↓
DISTINCT
↓
SELECT

- set = []
  for row in answer:
  set.add (row),
- ret / Print set

output
↓
[ G, 2008
PG, 2006 ]




NOT ( Money <= 50 AND Age ≤ 30 )

NOT Money ≤ = 50 OR NOT Age ≤ 30

T T