

```

4 // Part A
5 -- 1. How many directors are these per nationality?
6 select nationality,count(nationality) as count from directors
7 group by nationality
8 order by count desc
9
10 -- 2. What is the sum total movie length for each age certificate?
11 select age_certificate,movie_lang,sum(movie_length) from movies
12 group by age_certificate,movie_lang
13
14 -- 3. Return the movie languages which have a sum total movie length
15 select movie_lang, sum(movie_length) as total from movies
16 group by movie_lang
17 having sum(movie_length) > 500
18
19 // Part B
20 -- 1. Select the directors first and last names, the movie name
21 select dr.first_name, dr.last_name, mo.movie_name, mo.release_date
22 from directors dr join movies mo
23 on dr.director_id = mo.director_id
24 group by dr.first_name, dr.last_name, mo.movie_name, mo.release_date
25 having mo.movie_lang in ('Chinese' 'Japanese' 'Korean')

```

Data Output Explain Messages Notifications

	nationality character varying (20)		count bigint	
1	American		16	
2	British		6	
3	Chinese		4	
4	Japanese		3	
5	Brazilian		2	
6	Swedish		1	
7	South Korean		1	
8	Australian		1	
9	German		1	
10	Mexican		1	
11	French		1	

```

9
10 -- 2. What is the sum total movie length for each age certificate and movie language combination?
11 select age_certificate, movie_lang, sum(movie_length) from movies
12 group by age_certificate, movie_lang
13
14 -- 3. Return the movie languages which have a sum total movie length of over 500 minutes.
15 select movie_lang, sum(movie_length) as total from movies
16 group by movie_lang
17 having sum(movie_length) > 500
18
19 // Part B
20 -- 1. Select the directors first and last names, the movie names and release dates for all Chinese, K
21 select dr.first_name, dr.last_name, mo.movie_name, mo.release_date
22 from directors dr join movies mo
23 on dr.director_id = mo.director_id
24

```

Data Output Explain Messages Notifications

	age_certificate character varying (5)	movie_lang character varying (20)	sum bigint
1	15	Swedish	128
2	PG	English	1364
3	18	Portuguese	145
4	PG	Spanish	98
5	18	Korean	130
6	18	Japanese	219
7	15	Chinese	113
8	15	Portuguese	140
9	U	English	393
10	12	English	929
11	U	Japanese	227
12	18	English	500
13	15	German	165
14	12	Chinese	496
15	15	English	1638

```
14 -- 3. Return the movie languages which have a sum total movie length of over 500 minutes.  
15 select movie_lang, sum(movie_length) as total from movies  
16 group by movie_lang  
17 having sum(movie_length) > 500  
18  
19 // Part B  
20 -- 1. Select the directors first and last names, the movie names and release dates for all C  
21 select dr.first_name, dr.last_name, mo.movie_name, mo.release_date  
22 from directors dr join movies mo  
23 on dr.director_id = mo.director_id  
24
```

Data Output Explain Messages Notifications

	movie_lang	total
	character varying (20)	bigint
1	Chinese	609
2	English	4824

```

19 // Part B
20 -- 1. Select the directors first and last names, the movie names and release dates for all Chinese, Korean and Japanese movies.
21 select dr.first_name, dr.last_name, mo.movie_name, mo.release_date
22 from directors dr join movies mo
23 on dr.director_id = mo.director_id
24 group by dr.first_name, dr.last_name, mo.movie_name, mo.release_date, mo.movie_lang
25 having mo.movie_lang in ('Chinese', 'Japanese', 'Korean')
26

```

Data Output Explain Messages Notifications

	<b>first_name</b> character varying (30) 	<b>last_name</b> character varying (30) 	<b>movie_name</b> character varying (50) 	<b>release_date</b> date 	
1	Ang	Lee	Crouching Tiger Hidden Dragon	2000-07-06	
2	Bruce	Lee	Way of the Dragon	1972-06-01	
3	Chan-wook	Park	Oldboy	2005-03-25	
4	Hayao	Miyazaki	Ponyo	2009-08-14	
5	Hayao	Miyazaki	Spirited Away	2001-06-19	
6	Kar Wai	Wong	Chungking Express	1996-08-03	
7	Kar Wai	Wong	In the Mood for Love	2001-02-02	
8	Kinji	Fukasaku	Battle Royale	2001-01-04	
9	Sion	Sono	Cold Fish	2010-09-12	
10	Yimou	Zhang	House of Flying Daggers	2004-03-12	

```

28 -- 2. Select the movie names, release dates and international takings of all English language movies.
29 select mo.movie_name, mo.release_date, mr.international_takings
30 from movies mo join movie_revenues mr
31 on mo.movie_id = mr.movie_id
32 group by mo.movie_name, mo.release_date, mr.international_takings, mo.movie_lang
33 having mo.movie_lang in ('English') and mr.international_takings is not null
34 order by mr.international_takings desc
35

```

Data Output Explain Messages Notifications

	movie_name character varying (50)	release_date date	international_takings numeric (6,2)
1	Titanic	1997-12-19	1528.10
2	Spider-Man 3	2007-05-04	554.00
3	Life of Pi	2012-11-21	484.10
4	Spider-Man	2002-05-03	418.10
5	Spider-Man 2	2004-06-30	410.40
6	Forrest Gump	1994-07-06	348.10
7	Star Wars: A New Hope	1977-05-25	314.20
8	Gladiator	2000-05-05	273.40
9	Star Wars: Empire Strikes Back	1980-05-21	247.80
10	Jaws	1975-06-20	210.90
11	Gone with the Wind	1939-12-15	201.20
12	The Fifth Element	1997-05-09	200.30

```

37 -- 3. Select the movie names, domestic takings and international takings for all movies
38 -- with either missing domestic takings or missing international takings and order the results by movie name.
39 select mo.movie_name, mr.domestic_takings, mr.international_takings
40 from movies mo join movie_revenues mr
41 on mo.movie_id = mr.movie_id
42 group by mo.movie_name, mr.domestic_takings, mr.international_takings
43 having mr.domestic_takings is null or mr.international_takings is null
44 order by mo.movie_name
45

```

Data Output Explain Messages Notifications

	movie_name character varying (50)	domestic_takings numeric (6,2)	international_takings numeric (6,2)
1	A Clockwork Orange	27.10	[null]
2	Apocalypse Now	83.40	[null]
3	Battle Royale	[null]	[null]
4	Blade Runner	33.30	[null]
5	Chungking Express	[null]	[null]
6	Cold Fish	[null]	[null]
7	Goodfellas	46.60	[null]
8	Leon	[null]	[null]
9	Life of Brian	19.60	[null]
10	Mary Poppins	102.10	[null]
11	Raging Bull	23.10	[null]
12	Rushmore	16.90	[null]
13	Taxi Driver	28.10	[null]
14	The Shining	44.10	[null]
15	The Sound of Music	158.70	[null]
16	Way of the Dragon	[null]	[null]

```

46
47 -- 4. Count the number of movies that each director has directed.
48 select dr.first_name, dr.last_name, count(movie_name)
49 from directors dr join movies mo
50 on dr.director_id = mo.director_id
51 group by dr.first_name, dr.last_name
52 order by count(movie_name) desc
53

```

Data Output Explain Messages Notifications

	first_name character varying (30)	last_name character varying (30)	count bigint
1	George	Lucas	3
2	Martin	Scorsese	3
3	Wes	Anderson	3
4	Sam	Raimi	3
5	Stanley	Kubrick	3
6	Hayao	Miyazaki	2
7	Luc	Besson	2
8	Ang	Lee	2
9	Kar Wai	Wong	2
10	Victor	Fleming	2
11	Ridley	Scott	2
12	Robert	Stevenson	1
13	Steven	Spielberg	1
14	Richard	Ayoade	1
15	Paulo	Morelli	1
16	Kinji	Fukasaku	1
17	Zack	Snyder	1
18	Sion	Sono	1

```

54
55 -- 5 Select the first and last names of all the actors who have starred in the best international selling movie (or: in the movie directed by Wes Anderson)?
56 select ac.first_name, ac.last_name, mr.international_takings
57 from actors ac join movies_actors mac
58 on ac.actor_id = mac.actor_id
59 join movie_revenues mr
60 on mac.movie_id = mr.movie_id
61 group by ac.first_name, ac.last_name, mr.international_takings
62 having max(international_takings) is not null
63 order by max(international_takings) desc
64

```

Data Output Explain Messages Notifications

	first_name character varying (255)	last_name character varying (255)	international_takings numeric (6,2)
1	Sean	Young	1528.10
2	Patrick	Wilson	1528.10
3	[null]	Denden	1528.10
4	Jodie	Foster	554.00
5	Chris	Tucker	554.00
6	Aki	Maeda	554.00
7	Hiroki	Doi	554.00
8	Darlan	Cunha	554.00
9	Dandan	Song	484.10
10	Jason	Schwartzmann	484.10
11	Hei-Jung	Kang	484.10
12	Aki	Maeda	418.10
13	Hiroki	Doi	418.10
14	Jodie	Foster	418.10
15	Darlan	Cunha	418.10
16	Darlan	Cunha	410.40
17	Malcolm	McDowell	410.40
18	Hiroki	Doi	410.40

```
65 // Part C
66 -- 1. Select the first and last names of all the actors older than Marlon Brando
67 -- "Marlon" "Brando"      "1924-04-03"
68 select first_name, last_name, date_of_birth from actors
69 where date_of_birth < (select date_of_birth from actors
70 where first_name = 'Marlon' and last_name = 'Brando')
71 order by date_of_birth asc
```

Data Output Explain Messages Notifications

	first_name character varying (255)	last_name character varying (255)	date_of_birth date
1	Clark	Gable	1901-02-01
2	Scatman	Crothers	1910-05-23
3	Vivien	Leigh	1913-11-05
4	Alec	Guiness	1914-04-02
5	Judy	Garland	1922-06-10

```
73 -- 2. Select the movie names of all movies that have domestic takings above 300 million
74 select mo.movie_name
75 from movies mo join movie_revenues mr
76 on mo.movie_id = mr.movie_id
77 group by mo.movie_name, mr.domestic_takings
78 having mr.domestic_takings is not null and mr.domestic_takings > 300
79
```

Data Output Explain Messages Notifications

	movie_name character varying (50)	🔒
1	Spider-Man 2	
2	Titanic	
3	Spider-Man 3	
4	Star Wars: A New Hope	
5	Forrest Gump	
6	Star Wars: Return of the Jedi	
7	Spider-Man	

```
80 -- 3. Return the shortest and longest movie length for movies with an above average domestic takings.  
81 select min(movie_length) as shortest_movie_length,max(movie_length) as longest_movie_length  
82 from movies mo join movie_revenues mr  
83 on mo.movie_id = mr.movie_id  
84 where domestic_takings >  
85 (select avg(domestic_takings) from movie_revenues)  
86  
87
```

Data Output Explain Messages Notifications

	shortest_movie_length	longest_movie_length
	integer	integer
1	91	165

```
87  
88 -- 4. Select the first name, last name and date of birth for the oldest actors of each gender.  
89 select first_name, last_name, date_of_birth, gender  
90 from actors  
91 group by first_name, last_name, date_of_birth, gender  
92 having date_of_birth is not null and gender = 'M'  
93 order by date_of_birth asc  
94 limit 1  
95  
96  
97 select first_name, last_name, date_of_birth, gender  
98 from actors  
99 group by first_name, last_name, date_of_birth, gender  
100 having date_of_birth is not null and gender = 'F'  
101 order by date_of_birth asc  
102 limit 1  
103
```

Data Output Explain Messages Notifications

	first_name	last_name	date_of_birth	gender	
	character varying (255)	character varying (255)	date	character (1)	
1	Clark	Gable	1901-02-01	M	

```
87  
88 -- 4. Select the first name, last name and date of birth for the oldest actors of each gender.  
89 select first_name, last_name, date_of_birth,gender  
90 from actors  
91 group by first_name, last_name, date_of_birth, gender  
92 having date_of_birth is not null and gender = 'M'  
93 order by date_of_birth asc  
94 limit 1  
95  
96  
97 select first_name, last_name, date_of_birth,gender  
98 from actors  
99 group by first_name, last_name, date_of_birth, gender  
100 having date_of_birth is not null and gender = 'F'  
101 order by date_of_birth asc  
102 limit 1  
103
```

Data Output Explain Messages Notifications

	first_name	last_name	date_of_birth	gender	
	character varying (255)	character varying (255)	date	character (1)	
1	Vivien	Leigh	1913-11-05	F	

```

105
106 -- 5. Select the movie name, movie length and age certificate for movies with an above average length for their age certificate.
107 -- average length: 14.7428571428571429
108 select movie_name, movie_length, age_certificate
109 from movies
110 where age_certificate ~ '^-+?[0-9]*\.[0-9]+([eE]-+?[0-9]+)?$' and cast(age_certificate as integer)
111 >
112 (select avg(cast(age_certificate as integer)) from movies
113 WHERE age_certificate ~ '^-+?[0-9]*\.[0-9]+([eE]-+?[0-9]+)?$')
114 order by age_certificate desc
115

```

Data Output Explain Messages Notifications

	movie_name character varying (50)	movie_length integer	age_certificate character varying (5)
3	Eyes Wide Shut	130	18
4	City of God	145	18
5	Battle Royale	111	18
6	The Shining	126	18
7	Oldboy	130	18
8	A Clockwork Orange	112	18
9	Let the Right One In	128	15
10	Life of Brian	126	15
11	Never Let Me Go	117	15
12	Pulp Fiction	136	15
13	Submarine	115	15
14	Taxi Driver	117	15
15	The Lives of Others	165	15
16	There Will Be Blood	168	15
17	Three Billboards Outside Ebbi...	134	15