

1.- De la tabla videogames\_games, ordenar por año y mostrar resultados del nombre del juego que fue publicado por Nintendo en el año 1995 en adelante, que no se llame “Donkey Konga 2”, que vendiera entre cero y 0,32 en Europa, entre cero y 0,31 en America, y entre cero y 0,3 en otros territorios.

```
SELECT year,  
        name,  
        publisher, na_sales,eu_sales,other_sales  
from videogames_games  
        WHERE publisher = 'Nintendo' and year >= 1995 and NOT name = 'Donkey Konga 2'  
        and eu_sales BETWEEN 0 AND 0.32 and na_sales BETWEEN 0 and 0.31  
        and other_sales BETWEEN 0 and 0.3  
order by year  
limit 1
```

```
1 SELECT year,  
2     name,  
3     publisher, na_sales,eu_sales,other_sales  
4 FROM videogames_games  
5     WHERE publisher = 'Nintendo' AND year >= 1995 AND NOT name = 'Donkey Konga 2'  
6         AND eu_sales BETWEEN 0 AND 0.32 AND na_sales BETWEEN 0 AND 0.31  
7         AND other_sales BETWEEN 0 AND 0.3  
8 ORDER BY year  
9 LIMIT 1
```

2.- Tabla: star\_wars\_characters. Nombre de los personajes de Star Wars que tengan los ojos distintos a rojo y que sean de los mundos Chandrila, Stewjon o Tatooine. Añadir un nuevo campo “tamaño” que nos diga si es “grande” cuando la altura sea mayor o igual a 200 y “pequeño” si es menor de 200. Por último ordenar por altura de menor a mayor.

```
SELECT name, eye_color, homeworld, height,
```

```
case when height >=200 then 'grande'
```

```
    else 'pequeño'
```

```
end as 'tamaño'
```

```
FROM star_wars_characters_2
```

```
WHERE eye_color<> 'red' and homeworld in ('Chandrila','Stewjon','Tatooine')
```

```
ORDER BY height
```

```
10 SELECT name, eye_color, homeworld, height,
11 CASE WHEN height >=200 THEN 'grande'
12     ELSE 'pequeño'
13     END AS 'tamaño'
14
15 FROM star_wars_characters_2
16 WHERE eye_color<> 'red' AND homeworld IN ('Chandrila','Stewjon','Tatooine')
17 ORDER BY height
```

3.- De la tabla fortune, queremos ver sólo las 100 primeras compañías según el rango, sin incluir a Citigroup, viendo sólo las compañías dentro del sector financiero; industria diversified financials y crearemos un nuevo campo de rentabilidad en el que divida a las compañías entre rentables y no rentables si sus beneficios son mayores o menores a 1000.

```
SELECT company,industry,revenue,  
case when profits>1000 then 'rentables'  
      else 'no_rentables'  
      end as rentabilidad  
  
from fortune  
  
where company <> 'Citigroup'and industry='Diversified Financials'  
  
limit 100
```

```
9 SELECT company,industry,revenue,  
10 CASE WHEN profits>1000 THEN 'rentables'  
11     ELSE 'no_rentables'  
12     END AS rentabilidad  
13  
14 FROM fortune  
15 WHERE company <> 'Citigroup'AND industry='Diversified Financials'  
16 LIMIT 100
```

4.- En la tabla gobierno\_paro, filtrar por el mes de 'Enero de 2018' y el paro de hombres y mujeres de entre 25 y 45 años entre 1100 y 20 y ordenar los municipios con esos filtros de mayor paro resgistrado (del total) a menor. Por

último, añadir una nueva columna llamada 'Comparación de paro', agrupando el número total de parados en: mucho si es mayor que 1000, medio si es mayor que 500 y poco si es por debajo o igual a 500.

```
SELECT mes,provincia,  
case when total_paro_registrado >1000 then 'mucho'  
      when total_paro_registrado >500 then 'medio'  
      when total_paro_registrado <=500 then 'poco'  
      end as Comparacion_de_paro  
FROM gobierno_paro  
where mes='Enero de 2018'  
      and paro_hombre_edad_25__45 BETWEEN 20 and 1100  
      and paro_mujer_edad_25__45 BETWEEN 20 and 1100
```

```
9 SELECT mes,provincia,  
10 CASE WHEN total_paro_registrado >1000 THEN 'mucho'  
11     WHEN total_paro_registrado >500 THEN 'medio'  
12     WHEN total_paro_registrado <=500 THEN 'poco'  
13     END AS Comparacion_de_paro  
14 FROM gobierno_paro  
15 WHERE mes='Enero de 2018'  
16     AND paro_hombre_edad_25__45 BETWEEN 20 AND 1100  
17     AND paro_mujer_edad_25__45 BETWEEN 20 AND 1100  
18 |
```

5.- En la tabla star\_wars\_characters, buscar el personaje que sea de un planeta que comience por la letra T y que no termine por la letra d, que mida más de 200 y que no sea un droide. Pista: el caracter comodin es %.

SELECT name,homeworld, height,species

from star\_wars\_characters\_2

where homeworld like 'T%'AND homeworld NOT LIKE '%d'and height >200 and species <> 'droid'

```
7 SELECT name,homeworld, height,species
8 FROM star_wars_characters_2
9 WHERE homeworld LIKE 'T%'AND homeworld NOT LIKE '%d'AND height >200 AND species <> 'droid'
```

6.- En la tabla videogames\_games, queremos ver los nombres de los juegos de platform\_code 'DS' y años ordenados por año, con las ventas en las tres regiones principales (NA, EU y JP).

Teniendo en cuenta sólo estas regiones (ignorar other\_sales), queremos determinar en qué región ha vendido mejor cada juego. Para ello, generar un

campo nuevo best\_selling\_region cuyo valor sea 'NA', si NA\_sales es mayor que EU\_sales y que JP\_sales; 'EU' en caso de que EU\_sales sea mayor que NA\_sales y JP\_sales o 'JP' en el el tercer caso.

```
select name, year,na_sales,eu_sales,jp_sales,  
CASE WHEN na_sales>eu_sales and na_sales>jp_sales then 'na'  
      WHEN eu_sales>na_sales and eu_sales>jp_sales then 'eu'  
      else 'jp'  
  
end as best_selling_region  
FROM videogames_games  
order by year
```

```
11 SELECT name, year,na_sales,eu_sales,jp_sales,  
12 CASE WHEN na_sales>eu_sales AND na_sales>jp_sales THEN 'na'  
13      WHEN eu_sales>na_sales AND eu_sales>jp_sales THEN 'eu'  
14      ELSE 'jp'  
15  
16 END AS best_selling_region  
17 FROM videogames_games|  
18 ORDER BY year
```

7.- Cuales son las películas en USA, ordenados de forma descendente por duración en la tabla de imdb\_movies. añadir el nuevo campo "SON MUY LARGAS" a las que duren de 120 minutos a 200 minutos y "son cortas" las que duren menos 120.

```
SELECT movie_title,country,
```

case when duration between 120 and 200 then 'son muy largas'  
when duration <120 then 'son muy cortas'

end as son\_muy\_largas

from imdb\_movies

where country='USA'

order by duration DESC

```
6 SELECT movie_title, country,  
7 CASE WHEN duration BETWEEN 120 AND 200 THEN 'son muy largas'  
8     WHEN duration <120 THEN 'son muy cortas'  
9  
10 END AS son_muy_largas  
11 FROM imdb_movies  
12 WHERE country='USA'  
13 ORDER BY duration DESC
```

8.- De la tabla gobierno\_paro, cuál es la cifra de mujeres de 45 años que estaban en paro, donde la cifra de parados sin empleo anterior era cero, durante el mes de marzo de 2018, ordenado por municipios por orden alfabético descendente.

SELECT paro\_mujer\_edad\_\_\_\_45, paro\_sin\_empleo\_anterior, mes

from gobierno\_paro

where paro\_sin\_empleo\_anterior=0 and mes= 'Marzo de 2018'

order by municipio DESC

```
7 SELECT paro_mujer_edad__45, paro_sin_empleo_anterior, mes
8
9 FROM gobierno_paro
10 WHERE paro_sin_empleo_anterior=0 AND mes= 'Marzo de 2018'
11 ORDER BY municipio DESC
```

9.- Las 50 películas a color de USA, desde el año 2000 a 2010 con un campo nuevo llamado 'Calificación según Facebook' donde se evidencien tres variables: si tiene menos de 5.000 likes es 'Regular'; si tiene entre 5.000 y 10.000 likes es 'Buena' y si tiene más de 10.000 likes es 'Muy Buena'.

SELECT movie\_title, color, cast\_total\_facebook\_likes,

case when cast\_total\_facebook\_likes <5000 then 'regular'



```

when cast_total_facebook_likes BETWEEN 5000 and 10000 then 'buena'
when cast_total_facebook_likes > 10000 then 'muy buena'
end as Calificación_según_Facebook
from imdb_movies
where color= 'Color'
limit 50

```

```

8 SELECT movie_title, color, cast_total_facebook_likes,
9 CASE WHEN cast_total_facebook_likes <5000 THEN 'regular'
10 WHEN cast_total_facebook_likes BETWEEN 5000 AND 10000 THEN 'buena'
11 WHEN cast_total_facebook_likes > 10000 THEN 'muy buena'
12 END AS Calificación_según_Facebook
13 FROM imdb_movies
14 WHERE color= 'Color'
15 LIMIT 50

```

## MI QUERY

De la tabla imdb\_movies

Cuales son las 10 películas que tienen mayor puntuación de imdb, y de esas 10 cuáles son las de mayor número de votos de los usuarios.

Visualizar solo las que no sean western.

Crea un campo que clasifique las películas por su número de votos de usuarios que se llame “votos” donde si tiene más de 7 se llamen top, si tienen entre 5 y 7 se llamen se puede ver y si es menos que se llame no pierdas el tiempo.

```

SELECT movie_title, imdb_score,num_voted_users,gender,
case when num_voted_users >7 then 'top'
      when num_voted_users BETWEEN 5 and 7 then 'se pueden ver'
      else 'no pierdas el tiempo'
end as votos
FROM imdb_movies

```

where gender not like '%Western%'

order by imdb\_score desc, num\_voted\_users DESC

limit 10

```
1 SELECT movie_title, imdb_score,num_voted_users,gender,
2 CASE WHEN num_voted_users >7 THEN 'top'
3     WHEN num_voted_users BETWEEN 5 AND 7 THEN 'se pueden ver'
4     ELSE 'no pierdas el tiempo'
5     END AS votos
6
7 FROM imdb_movies
8 WHERE gender NOT LIKE '%Western%'
9 ORDER BY imdb_score DESC, num_voted_users DESC
10
11 LIMIT 10
```

movie_title	imdb_score	num_voted_us...	gender	votos
Towering Inferno	9.5	10	Comedy	top
Kickboxer: Veng...	9.1	246	Action	top
The Matrix	8.7	1217752	Action Sci-Fi	top
The Honeymoon...	8.7	3446	Comedy Family	top
Butterfly Girl	8.7	27	Documentary	top
A Beginner's Gui...	8.7	13	Comedy Horror ...	top
Modern Times	8.6	143086	Comedy Drama ...	top
Running Forever	8.6	8	Family	top
Gladiator	8.5	982637	Action Drama R...	top
The Green Mile	8.5	782610	Crime Drama Fa...	top

## CARLOS

--Ordenar todos los álbumes que contengan la letra W

--y se sacaran entre los años 1975 y 2001

--según la paciencia del público para ello crear un campo que se denomine  
--Paciencia del público en el que los álbumes con más duración de 50 minutos  
sea "ADECUADO",  
--entre 51 y 70 sea "SOPORTABLE", más de 70 sea "INSOPORTABLE" y lo  
demás "IDEAL".

```
SELECT album, release_year,duration,minutes,  
case when minutes >70 then ' insoportable'  
      WHEN minutes = 50 then 'adecuado'  
      when minutes BETWEEN 51 and 70 then 'soportable'  
      else 'ideal'  
end as paciencia_del_publico  
from rolling_top_albums_1  
where release_year BETWEEN 1975 and 2001 and album like '%w%'  
order by paciencia_del_publico
```

```
7 SELECT album, release_year,duration,minutes,  
8 CASE WHEN minutes >70 THEN ' insoportable'  
9     WHEN minutes = 50 THEN 'adecuado'  
10    WHEN minutes BETWEEN 51 AND 70 THEN 'soportable'  
11    ELSE 'ideal'  
12 END AS paciencia_del_publico  
13 FROM rolling_top_albums_1  
14 WHERE release_year BETWEEN 1975 AND 2001 AND album LIKE '%w%'  
15 ORDER BY paciencia_del_publico
```

## ARTURO

--De la tabla starstar\_wars\_characters\_2.  
--Mostrar los personajes que sean humanos, hutt, wookie o ewok  
--con peso no nulo ordenando por peso de mayor a menor.

--Añadir un nuevo campo 'Recomendación de la abuela'  
--dónde se indique que si pesan más de 1000 necesitan comer más lechuga,  
--si su peso está entre 90 y 1000 están fuertes  
--y si pesan menos de 90 necesitan comer más.

```
SELECT name,species,mass,  
case WHEN mass >1000 then 'necesitan comer más lechuga'  
      when mass BETWEEN 90 and 1000 then 'están fuertes'  
      when mass < 90 then 'necesitan comer más'  
end as recomendacion_de_la_abuela
```

```
from star_wars_characters_2  
      where mass notnull and species in('Human','Hutt','Wookie','Ewok')  
      order by mass DESC
```

```
9 SELECT name,species,mass,  
10 CASE WHEN mass >1000 THEN 'necesitan comer más lechuga'  
11     WHEN mass BETWEEN 90 AND 1000 THEN 'están fuertes'  
12     WHEN mass < 90 THEN 'necesitan comer más'  
13     END AS recomendacion_de_la_abuela  
14  
15 FROM star_wars_characters_2  
16     WHERE mass NOTNULL AND species IN('Human','Hutt','Wookie','Ewok')  
17     ORDER BY mass DESC
```

## VICTOR

--De la tabla fortune, mostrar las Empresas Medianas ordenadas por ranking,  
--siguiendo los siguientes criterios de clasificación según número de  
empleados y beneficios

--(filtrando los valores nulos):

--Si la empresa tiene más de 500 empleados, es una Empresa Grande

--Si la empresa tiene entre 50 y 500 empleados, es una Empresa Mediana

--Si la empresa tiene menos de 50 empleados, es una Empresa Pequeña

--Si la empresa tiene beneficios, es Con Beneficios

--Si la empresa NO tiene beneficios, es Con Pérdidas.

SELECT rank,employees,profits,

CASE WHEN employees >500 THEN 'Empresa Grande'

when employees BETWEEN 50 and 500 then 'Empresa Mediana'

WHEN employees <50 then 'Empresa Pequeña'

end as empleados,

CASE when profits >0 then 'Con Beneficios'

when profits<0 then 'Con Pérdidas'

end as beneficios

FROM fortune

WHERE employees notnull and profits NOTNULL and empleados='Empresa Mediana'

order by rank

```

11 SELECT rank,employees,profits,
12 CASE WHEN employees >500 THEN 'Empresa Grande'
13     WHEN employees BETWEEN 50 AND 500 THEN 'Empresa Mediana'
14     WHEN employees <50 THEN 'Empresa Pequeña'
15     END AS empleados,
16
17 CASE WHEN profits >0 THEN 'Con Beneficios'
18     WHEN profits<0 THEN 'Con Pérdidas'
19     END AS beneficios
20 FROM fortune
21 WHERE employees NOTNULL AND profits NOTNULL AND empleados='Empresa Mediana'
22 ORDER BY rank

```

JAVIER

--De la tabla videogames\_games,  
 --mostrar juegos de Playstation que el nombre no contenga "Jampack".  
 --Ordenar por año de mayor a menor.  
 --Teniendo en cuenta la editora,  
 --mostrar una columna nueva que indica si fue publicado por "Sony" o "otros".

```
SELECT name,year,publisher,
```

```

CASE WHEN publisher LIKE '%Sony%' then 'Sony'
      else 'otros'

```

```
end as editora
```

```
FROM videogames_games
```

```
WHERE platform_code='PS'AND name not like ('%Jampack%')
```

```
order by year desc
```

```

7  SELECT name,year,publisher,
8
9  CASE WHEN publisher LIKE '%Sony%' THEN 'Sony'
10         ELSE 'otros'
11  END AS editora
12  FROM videogames_games
13 WHERE platform_code='PS' AND name NOT LIKE ('%Jampack%')
14 ORDER BY year DESC

```

## ANDRÉ

- En la tabla videogames\_consoles,
- traer todas las consolas y clasificarlos en función de sus ventas,
- Si una consola ha vendido menos de 30 millones de unidades, se etiqueta como "Bajo".
- Si ha vendido entre 30 y 60 millones de unidades, se etiqueta como "Medio".
- Si ha vendido más de 60 millones de unidades, se etiqueta como "Alto",
- por último ordenar por el nombre de la consola alfabéticamente
- además de no incluir las consolas que empiecen por números.
- La consulta devuelve el nombre de la consola y su clasificación de ventas correspondiente.

```

select console_name,units_sold_million,
CASE WHEN units_sold_million < 30 then 'Bajo'
      when units_sold_million BETWEEN 30 and 60 then 'Medio'
      when units_sold_million > 60 then 'Alto'
      end as ventas

from videogames_consoles

where console_name not like '0%' AND console_name not like '1%' and
      console_name not like '2%' AND console_name not like '3%'and
console_name not like '4%'

```

AND console\_name not like '5%' AND console\_name not like '6%' AND console\_name not like '7%'

AND console\_name not like '8%' AND console\_name not like '9%'

order by console\_name

```
10 SELECT console_name,units_sold_million,
11 CASE WHEN units_sold_million < 30 THEN 'Bajo'
12      WHEN units_sold_million BETWEEN 30 AND 60 THEN 'Medio'
13      WHEN units_sold_million > 60 THEN 'Alto'
14      END AS ventas
15
16 FROM videogames_consoles
17 WHERE console_name NOT LIKE '0%' AND console_name NOT LIKE '1%' AND
18       console_name NOT LIKE '2%' AND console_name NOT LIKE '3%' AND console_name NOT LIKE '4%'
19       AND console_name NOT LIKE '5%' AND console_name NOT LIKE '6%' AND console_name NOT LIKE '7%'
20       AND console_name NOT LIKE '8%' AND console_name NOT LIKE '9%'
21 ORDER BY console_name
```

## ESTEBAN

--De la tabla gobierno\_paro mostrar la cantidad de paro en los

--sectores agricultura, industria, construcción y servicios en el mes Enero

--ordenados por municipio,

--creando un campo comparacion\_de\_paro

--indicando si el total es mayor de 1000 como “alto” mayor de 500 “medio” y menor de 500 “bajo”

SELECT paro\_agricultura,paro\_industria,paro\_construccion,paro\_servicios,

case when total\_paro\_registrado >1000 then 'alto'

when total\_paro\_registrado >500 then 'medio'

when total\_paro\_registrado <500 THEN 'bajo'

end as comparacion\_de\_paro

from gobierno\_paro



where mes LIKE 'Enero%'

order by municipio

```
7 SELECT paro_agricultura,paro_industria,paro_construcci__n,paro_servicios,  
8 CASE WHEN total_paro_registrado >1000 THEN 'alto'  
9     WHEN total_paro_registrado >500 THEN 'medio'  
10    WHEN total_paro_registrado <500 THEN 'bajo'  
11    END AS comparacion_de_paro  
12 FROM gobierno_paro  
13 WHERE mes LIKE 'Enero%'  
14 ORDER BY municipio|
```

ROBERTO

--Lista de personajes de Star Wars sin contar a los rubios,

--que no midan más de 200 de altura y no sean naturales de Kashyyyk.

--Aparte, crear un nuevo campo que especifique su alineamiento según el color de sus ojos

--(azul = bueno, rojo = malo).

--Partiendo de la idea de que todos los droides son neutrales

--sin importar el color de sus ojos.

select name, hair\_color,eye\_color, height,

case when eye\_color ='blue' then 'bueno'

when eye\_color = 'red' then 'malo'

end as alineamiento

from star\_wars\_characters\_2

WHERE hair\_color <>'blond' and height <=200 and homeworld <>'Kashyyyk'

order by alineamiento

```

8 SELECT name, hair_color, eye_color, height,
9 CASE WHEN eye_color = 'blue' THEN 'bueno'
10      WHEN eye_color = 'red' THEN 'malo'
11      END AS alineamiento
12      |
13 FROM star_wars_characters_2
14 WHERE hair_color <> 'blond' AND height <= 200 AND homeworld <> 'Kashyyyk'
15 ORDER BY alineamiento

```

## MARCOS

--En la tabla StarWars ,

--qué personajes femeninos de Tatooine existen cuyo tipo de pelo comience con la letra “b”

--y tengan altura superior a 140 ?

--(Crea adicionalmente una columna que indique “Mujeres de Tatooine”

-- en los casos en los que aplique y “Otros” para el resto) .

```

SELECT name, gender, homeworld, hair_color, height,
case when homeworld = 'Tatooine' then 'Tatooine'
      else 'otros'

```

```

end as mujeres_de_tatooine

```

```

from star_wars_characters_2

```

```

where gender = 'female' and homeworld = 'Tatooine'

```

```

and hair_color like 'b%' and hair_color like 'B%' and height > 140

```

```

SELECT name,gender,homeworld, hair_color,height,
CASE WHEN homeworld = 'Tatooine' THEN 'Tatooine'
      ELSE 'otros'
END AS mujeres_de_tatooine
FROM star_wars_characters_2
WHERE gender ='female' AND homeworld ='Tatooine'
AND hair_color LIKE 'b%' AND hair_color LIKE 'B%' AND height>140

```

### CRISTIAN

De la tabla Loan-Data filtrar los clientes dueños de su propio hogar, que posean ahorros y sean altamente calificados laboralmente. Crear la columna 'Potencial Hipotecario' donde según el nivel de ahorros sea 'Bajo' cuando sus ahorros sean inferiores a 100, 'Medio' cuando sus ahorros sean entre 100 y 500, 'Alto' cuando sus ahorros sean entre 500 y 1000 y 'Muy Alto' cuando sus ahorros sean mayor o igual a 1000.

```

--De la tabla Loan-Data filtrar los clientes dueños de su propio hogar,
--que posean ahorros y sean altamente calificados laboralmente.
--Crear la columna 'Potencial Hipotecario' donde según el nivel de ahorros sea
--'Bajo'cuando sus ahorros sean inferiores a 100,
--'Medio' cuando sus ahorros sean entre 100 y 500,
--'Alto' cuando sus ahorros sean entre 500 y 1000 y
--'Muy Alto' cuando sus ahorros sean mayor o igual a 1000.

```

```

SELECT job, savings_status,housing,
CASE WHEN savings_status = '<100' then 'bajo'
      when savings_status = '100<=X<500' then 'medio'
      when savings_status ='500<=X<1000' then 'alto'
      when savings_status ='>=1000' then 'muy alto'
      end as Potencial_Hipotecario
FROM LoanData
WHERE savings_status <> 'no known savings' and job ='high qualif/self emp/mgmt' and housing
='own'

```

```

9 SELECT job, savings_status,housing,
10 CASE WHEN savings_status = '<100' THEN 'bajo'
11      WHEN savings_status = '100<=X<500' THEN 'medio'
12      WHEN savings_status ='500<=X<1000' THEN 'alto'
13      WHEN savings_status ='>=1000' THEN 'muy alto'
14      END AS Potencial_Hipotecario
15 FROM LoanData
16 WHERE savings_status <> 'no known savings' AND job ='high qualif/self emp/mgmt' AND housing ='own'

```

## María otero

--En la tabla star\_wars\_characters,

--Establecer los personajes femeninos humanos ordenados por el nombre que no pertenezcan a Naboo y no tengan los ojos azules.

--Establecer una nueva columna "COINCIDENCIAS" donde si coinciden el color de pelo y de ojos indique "iguales" y en caso contrario indique "distintos"

--(pista, solo pueden quedar tres)

select name, species, homeworld,gender,eye\_color,hair\_color,

case when eye\_color=hair\_color then 'iguales'

ELSE 'distintos'

end as coincidencias

from star\_wars\_characters\_2

where species='Human' and gender ='female' and homeworld <> 'Naboo' and eye\_color <>'blue'

order by name

```
5 SELECT name, species, homeworld,gender,eye_color,hair_color,
6 CASE WHEN eye_color=hair_color THEN 'iguales'
7     ELSE 'distintos'
8     END AS coincidencias
9 FROM star_wars_characters_2
10 WHERE species='Human' AND gender ='female' AND homeworld <> 'Naboo' AND eye_color <>'blue'
11 ORDER BY name
```

## DIEGO

Tabla, world\_health\_org

ordena alfabéticamente de forma descendente los 5 países que cumplan:

- Pertenezcan a EN VIAS DE DESARROLLO
- más de 30 millones de habitantes
- que no acabe en "a"

Para ello se debe crear un nuevo campo llamado grado\_desarrollo que se asigne del siguiente modo:

- EN VIAS DE DESARROLLO: PIB <5000 \$ y tasa de fertilidad > 50
- DESARROLLADOS: PIB > 10000 \$ y tasa de fertilidad <20

--Tabla, world\_health\_org

--ordena alfabéticamente de forma descendente los 5 países que cumplan:

-- Pertenezcan a EN VIAS DE DESARROLLO

-- más de 30 millones de habitantes

-- que no acabe en "a"

--Para ello se debe crear un nuevo campo llamado grado\_desarrollo que se asigne del siguiente modo:

-- EN VIAS DE DESARROLLO: PIB <5000 \$ y tasa de fertilidad > 50

-- DESARROLLADOS: PIB > 10000 \$ y tasa de fertilidad <20

SELECT country,gross\_income\_per\_capita, adolescent\_fertility\_rate,

case when gross\_income\_per\_capita <5000 and adolescent\_fertility\_rate >50 then 'en vias de desarrollo'

when gross\_income\_per\_capita >10000 and adolescent\_fertility\_rate<20 then 'desarrollados'

end as grado\_desarrollo

FROM world\_health\_org

where grado\_desarrollo= 'en vias de desarrollo' and population\_in\_thousands >30 and country NOT like '%a'

order by country desc

```

11 SELECT country,gross_income_per_capita, adolescent_fertility_rate,
12 CASE WHEN gross_income_per_capita <5000 AND adolescent_fertility_rate >50 THEN 'en vias de desarrollo'
13      WHEN gross_income_per_capita >10000 AND adolescent_fertility_rate<20 THEN 'desarrollados'
14
15 END AS grado_desarrollo
16
17 FROM world_health_org
18 WHERE grado_desarrollo= 'en vias de desarrollo' AND population_in_thousands >30 AND country NOT LIKE '%a'
19 ORDER BY country DESC

```

## DAVID

En la tabla 'fortune' mostrar todas las compañías que pertenezcan al sector chemicals y que estén localizados en los puestos entre 700 y 900 del ranking ordenados de forma descendente. Crear un campo número\_empleados en el cual si el número de empleados es mayor de 6000 se añadirá un campo que ponga 'muchos empleados' y, en caso contrario, pondrá 'pocos empleados'.

--En la tabla 'fortune' mostrar todas las compañías que pertenezcan al sector chemicals

--y que estén localizados en los puestos entre 700 y 900 del ranking ordenados de forma descendente.

--Crear un campo número\_empleados en el cual si el número de empleados es mayor de 6000

--se añadirá un campo que ponga 'muchos empleados' y, en caso contrario, pondrá 'pocos empleados'.

```
select company,sector, rank,employees,
```

```
CASE when employees>6000 then 'muchos empleados'
```

```
else 'pocos empleados'
```

```
end as numero_empleados
```

```
from fortune
```

where sector='Chemicals' and rank BETWEEN 700 and 900

order by rank DESC

```
5 SELECT company,sector, rank,employees,
6 CASE WHEN employees>6000 THEN 'muchos empleados'
7 ELSE 'pocos empleados'
8
9 END AS numero_empleados
10
11
12 FROM fortune
13 WHERE sector='Chemicals' AND rank BETWEEN 700 AND 900
14 ORDER BY rank DESC
```

## MARÍA OTERO

En la tabla star\_wars\_characters,

Establecer los personajes femeninos humanos ordenados por el nombre que no pertenezcan a Naboo y no tengan los ojos azules.

Establecer una nueva columna “COINCIDENCIAS” donde si coinciden el color de pelo y de ojos indique “iguales” y en caso contrario indique “distintos”

(pista, solo pueden quedar tres)

## Jabson

De la tabla Loan-Data, se desea obtener la lista de personas que tengan un Crédito con una duración mayor o igual a 12 meses, que haya sido usado para comprar un “coche nuevo”, que sea “hombre soltero” y cuya edad sea mayor a 25 años. Se requiere además que cuando el monto de crédito sea mayor o igual a 6000 diga “Buen Crédito”, cuando sea mayor a 4000 diga “Crédito Aceptable” y cuando sea mayor a 2000 diga “Crédito Bajo” en un campo nuevo llamado “Tipo de Crédito”.

--De la tabla Loan-Data, se desea obtener la lista de personas que tengan

--un Crédito con una duración mayor o igual a 12 meses,

--que haya sido usado para comprar un “coche nuevo”,

--que sea “hombre soltero” y cuya edad sea mayor a 25 años.

--Se requiere además que cuando el monto de crédito sea mayor o igual a 6000 diga “Buen Crédito”,

--cuando sea mayor a 4000 diga “Crédito Aceptable”

--y cuando sea mayor a 2000 diga “Crédito Bajo” en un campo nuevo llamado “Tipo de Crédito”.

```
SELECT personal_status, purpose, duration, age, credit_amount,  
case when credit_amount >= 6000 then 'buen crédito'  
      when credit_amount > 4000 then 'credito aceptable'  
      when credit_amount > 2000 then 'credito bajo'  
end as tipo_de_credito
```

from LoanData

where duration >=12 and purpose ='new car' and age >25 and personal\_status>25

```
8 SELECT personal_status, purpose, duration, age, credit_amount,  
9 CASE WHEN credit_amount >= 6000 THEN 'buen crédito'  
10    WHEN credit_amount > 4000 THEN 'credito aceptable'  
11    WHEN credit_amount > 2000 THEN 'credito bajo'  
12    END AS tipo_de_credito  
13  
14 FROM LoanData  
15 WHERE duration >=12 AND purpose ='new car' AND age >25 AND personal_status>25
```

## Almudena

De la tabla stars\_war\_characters, clasificar como ‘Guapo’ a los que tenga pelo rubio y ojos azules. El resto serán ‘sin interés’ en una categoría denominada Clasificación.



Seleccionar los que tengan una altura entre 170 y 190 y sean humanos.

--De la tabla stars\_war\_characters,

--clasificar como 'Guapo' a los que tenga pelo rubio y ojos azules.

--El resto serán 'sin interés' en una categoría denominada Clasificación.

--Seleccionar los que tengan una altura entre 170 y 190 y sean humanos.

```
select name,eye_color,hair_color, height, species,  
case when eye_color='blue' and hair_color='blond' then 'guapo'  
else 'sin interés'  
end as clasificación
```

from star\_wars\_characters\_2

where height BETWEEN 170 and 190 and species='Human'

```
5  
6 SELECT name,eye_color,hair_color, height, species,  
7 CASE WHEN eye_color='blue' AND hair_color='blond' THEN 'guapo'  
8 ELSE 'sin interés'  
9 END AS clasificación  
10  
11 FROM star_wars_characters_2  
12 WHERE height BETWEEN 170 AND 190 AND species='Human'|
```

## Leticia

En la tabla jamesbond otórgale otro nombre a los campos Year y Film. Muestra el nombre de las películas que no estén dirigidas por Lewis Gilbert y se hayan estrenado entre 1960 y 2002. Añade un campo 'Presupuesto' donde si este es mayor o igual a 500 lo nombres como 'Presupuesto alto' y en el caso de ser menor a 500 como 'Presupuesto medio'. Ordena alfabéticamente el título de la película.

--En la tabla jamesbond otórgale otro nombre a los campos Year y Film.  
--Muestra el nombre de las películas que no estén dirigidas por Lewis Gilbert  
--y se hayan estrenado entre 1960 y 2002.  
--Añade un campo 'Presupuesto' donde si este es mayor o igual a 500 lo nombres como  
'Presupuesto alto'  
--y en el caso de ser menor a 500 como 'Presupuesto medio'. Ordena alfabéticamente el  
título de la película.

```
select year as año,film as pelicula, budget,  
case WHEN budget >=500 then 'presupuesto alto'  
      when budget < 500 then 'presupuesto medio'  
end as presupuesto  
from jamesbond  
where director <> 'Lewis Gilbert' and year BETWEEN 1960 and 2002
```

order by film

```
7 SELECT year AS año,film AS pelicula, budget,  
8 CASE WHEN budget >=500 THEN 'presupuesto alto'  
9      WHEN budget < 500 THEN 'presupuesto medio'  
10 END AS presupuesto  
11 FROM jamesbond  
12 WHERE director <> 'Lewis Gilbert' AND year BETWEEN 1960 AND 2002  
13  
14 ORDER BY film  
15 |
```

## LOURDES

Partiendo de la tabla star\_wars\_characters, crea una nueva columna denominada "personajes chachis", en la cual, los personajes que tengan el pelo de color brown, serán los molones y los que no tengan pelo, serán no molones.

Una vez determinados qué personajes son los que molan, buscar al mejor, en base a los siguientes parámetros:

- El nombre de su planeta de origen empieza por "k" y su nombre contenga la letra "w"

--Partiendo de la tabla star\_wars\_characters,

--crea una nueva columna denominada "personajes chachis",

--en la cual, los personajes que tengan el pelo de color brown, serán los molones y

--los que no tengan pelo, serán no molones.

--Una vez determinados qué personajes son los que molan, buscar al mejor,

--en base a los siguientes parámetros:

--El nombre de su planeta de origen empieza por “k” y su nombre contenga la letra “w”

```
SELECT name, hair_color, homeworld,  
case when hair_color='brown' then 'molones'  
else 'no molones'  
end as personajes_chachis  
from star_wars_characters_2  
where homeworld like 'k%' and name like '%w%' and personajes_chachis='molones'
```

```
9 SELECT name, hair_color, homeworld,  
10 CASE WHEN hair_color='brown' THEN 'molones'  
11 ELSE 'no molones'  
12 END AS personajes_chachis  
13 FROM star_wars_characters_2  
14 WHERE homeworld LIKE 'k%' AND name LIKE '%w%' AND personajes_chachis='molones'
```

## EMILIO

Ordena de menos a más los días del año 2017 que el bitcoin ha abierto por debajo de 3000.

Separa en otro grupo si el volumen de btc es mayor de 10000 como mercado btc y si el volumen de usd es mayor de 10 millones como mercado USD. Si no es ninguno marcalo como mercado dividido.

--Ordena de menos a más los días del año 2017 que el bitcoin ha abierto por debajo de 3000.

--Separa en otro grupo si el volumen de btc es mayor de 10000 como mercado btc  
--y si el volumen de usd es mayor de 10 millones como mercado USD.

--Si no es ninguno marcalo como mercado dividido.

```
select date, open, volume_btc,volume_usd,  
case when volume_btc<10000 then 'mercado btc'  
      when volume_usd>10000000 then 'mercado usd'  
      else 'mercado dividido'  
end as mercado  
from bitcoin_daily_rates_formatdate  
where date like '2017%' and open<3000  
order by date
```

```
6 SELECT date, OPEN, volume_btc,volume_usd,  
7 CASE WHEN volume_btc<10000 THEN 'mercado btc'  
8     WHEN volume_usd>10000000 THEN 'mercado usd'  
9     ELSE 'mercado dividido'  
10 END AS mercado  
11 FROM bitcoin_daily_rates_formatdate  
12 WHERE date LIKE '2017%' AND OPEN<3000  
13 ORDER BY date  
14
```