

INNER JOIN

TABLA UNO

ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5
1	NARANJA	6

SELECT *

FROM

TABLA_UNO AS TU

JOIN

TABLA_DOS AS TD

ON TU.KEY = TD.KEY

TABLA DOS

KEY	COLOR
1	AMARILLO
3	ROJO
5	ROSADO
7	MORADO
8	FUCSIA
9	SALMON

INNER JOIN

ID	COLOR	KEY	KEY	COLOR
6	AMARILLO	1	1	AMARILLO
5	ROJO	3	3	ROJO
3	ROSADO	5	5	ROSADO

SELF JOIN

TABLA UNO

ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5
1	NARANJA	6

SELECT *

FROM

TABLA_UNO AS TU_1

INNER JOIN

TABLA_UNO AS TU_2

ON TU_1.KEY = TU_2.ID

SELF JOIN

ID	COLOR	KEY	ID	COLOR	KEY
6	AMARILLO	1	1	NARANJA	6
4	AZUL	2	2	VERDE	4
5	ROJO	3	3	ROSADO	5
2	VERDE	4	4	AZUL	2
3	ROSADO	5	5	ROJO	3
1	NARANJA	6	6	AMARILLO	1

LEFT JOIN

TABLA UNO

ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5
1	NARANJA	6

SELECT *

FROM

TABLA_UNO AS TU

LEFT JOIN

TABLA_DOS AS TD

ON TU.KEY = TD.KEY

ORDER BY TD.KEY

TABLA DOS

KEY	COLOR
1	AMARILLO
3	ROJO
5	ROSADO
7	MORADO
8	FUCSIA
9	SALMON

LEFT JOIN

ID	COLOR	KEY	KEY	COLOR
6	AMARILLO	1	1	AMARILLO
5	ROJO	3	3	ROJO
3	ROSADO	5	5	ROSADO
4	AZUL	2	NULL	NULL
2	VERDE	4	NULL	NULL
1	NARANJA	6	NULL	NULL

RIGHT JOIN

TABLA UNO

ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5
1	NARANJA	6

SELECT *

FROM

TABLA_UNO AS TU

RIGHT JOIN

TABLA_DOS AS TD

ON TU.KEY = TD.KEY

ORDER BY TU.KEY

TABLA DOS

KEY	COLOR
1	AMARILLO
3	ROJO
5	ROSADO
7	MORADO
8	FUCSIA
9	SALMON

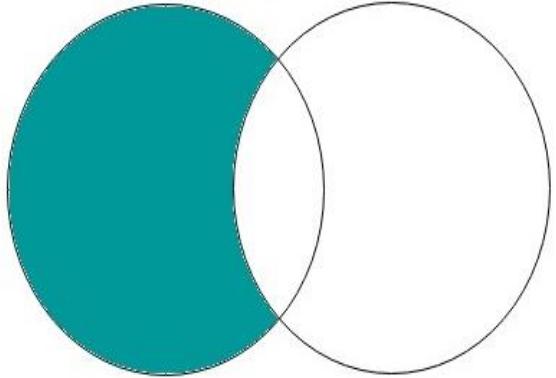
RIGHT JOIN

KEY	COLOR	ID	KEY	COLOR
1	AMARILLO	6	1	AMARILLO
3	ROJO	5	3	ROJO
5	ROSADO	3	5	ROSADO
NULL	NULL	NULL	7	MORADO
NULL	NULL	NULL	8	FUCSIA
NULL	NULL	NULL	9	SALMON

ANTI LEFT JOIN

TABLA UNO

ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5
1	NARANJA	6



SELECT *
FROM

LEFT JOIN

TABLA_UNO AS TU
ON TU.KEY = TD.KEY
WHERE TD.KEY IS NULL

TABLA DOS

KEY	COLOR
1	AMARILLO
3	ROJO
5	ROSADO
7	MORADO
8	FUCSIA
9	SALMON

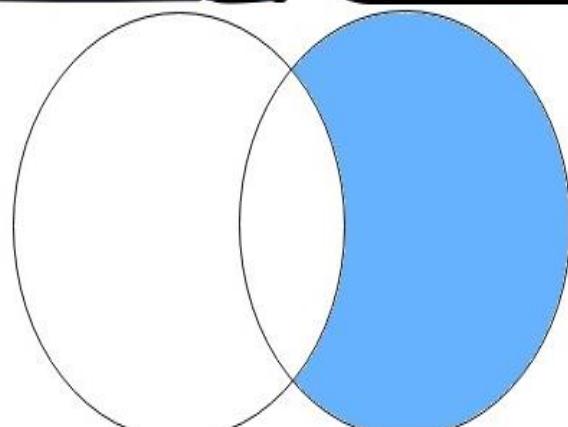
ANTI LEFT JOIN

ID	COLOR	KEY	KEY	COLOR
4	AZUL	2	NULL	NULL
2	VERDE	4	NULL	NULL
1	NARANJA	6	NULL	NULL

ANTI RIGHT JOIN

TABLA UNO

ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5
1	NARANJA	6



SELECT *

FROM

TABLA_UNO AS TU
RIGHT JOIN

TABLA_DOS AS TD
ON TU.KEY = TD.KEY
WHERE TU.KEY IS NULL

TABLA DOS

KEY	COLOR
1	AMARILLO
3	ROJO
5	ROSADO
7	MORADO
8	FUCSIA
9	SALMON

ANTI_RIGHT_JOIN

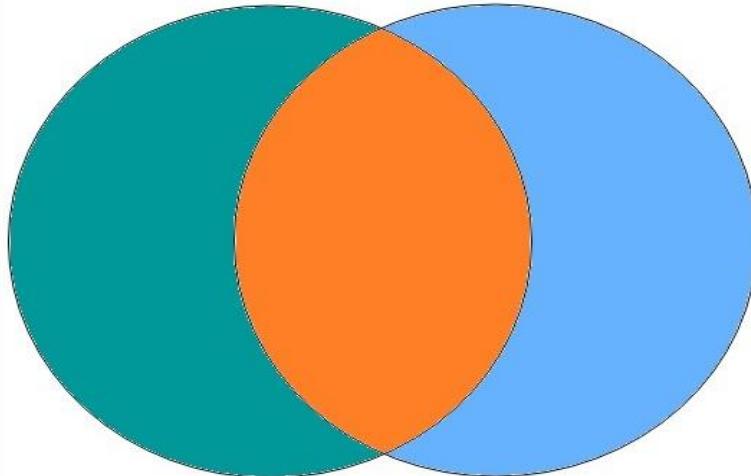
ID	COLOR	KEY	KEY	COLOR
NULL	NULL	NULL	7	MORADO
NULL	NULL	NULL	8	FUCSIA
NULL	NULL	NULL	9	SALMON

FULL JOIN

TABLA UNO		
ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5
1	NARANJA	6

SELECT *
FROM
TABLA_UNO AS TU
FULL JOIN
TABLA_DOS AS TD
ON TU.KEY = TD.KEY

TABLA DOS	
KEY	COLOR
1	AMARILLO
3	ROJO
5	ROSADO
7	MORADO
8	FUCSIA
9	SALMON



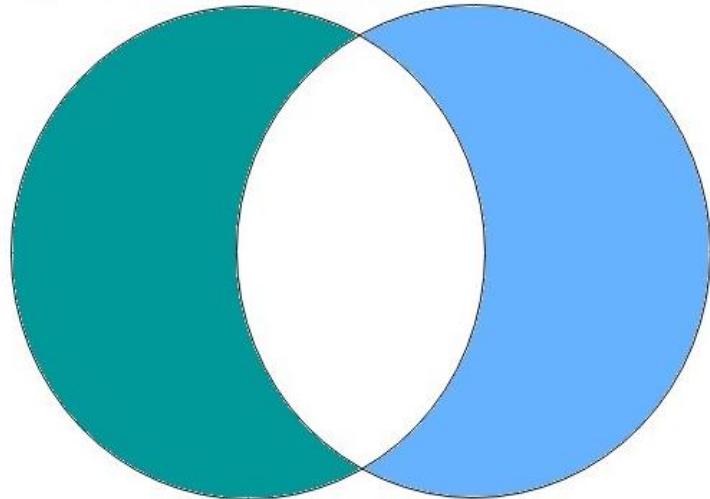
FULL JOIN				
ID	COLOR	KEY	KEY	COLOR
6	AMARILLO	1	1	AMARILLO
5	ROJO	3	3	ROJO
3	ROSADO	5	5	ROSADO
4	AZUL	2	NULL	NULL
2	VERDE	4	NULL	NULL
1	NARANJA	6	NULL	NULL
NULL	NULL	NULL	7	MORADO
NULL	NULL	NULL	8	FUCSIA
NULL	NULL	NULL	9	SALMON

ANTI FULL JOIN

TABLA UNO		
ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5
1	NARANJA	6

SELECT *
FROM
TABLA_UNO AS TU
FULL JOIN
TABLA_DOS AS TD
ON TU.KEY = TD.KEY
WHERE TU.KEY IS NULL
OR TD.KEY IS NULL

TABLA DOS	
KEY	COLOR
1	AMARILLO
3	ROJO
5	ROSADO
7	MORADO
8	FUCSIA
9	SALMON



ANTI_FULL JOIN				
ID	COLOR	KEY	KEY	COLOR
4	AZUL	2	NULL	NULL
2	VERDE	4	NULL	NULL
1	NARANJA	6	NULL	NULL
NULL	NULL	NULL	7	MORADO
NULL	NULL	NULL	8	FUCSIA
NULL	NULL	NULL	9	SALMON

CROSS JOIN

TABLA UNO		
ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5

SELECT *
FROM

TABLA_UNO AS TU
CROSS JOIN
TABLA_DOS AS TD

TABLA DOS	
KEY	COLOR
7	MORADO
8	FUCSIA
9	SALMON

CROSS JOIN

ID	COLOR	KEY	KEY	COLOR
6	AMARILLO	1	7	MORADO
4	AZUL	2	7	MORADO
5	ROJO	3	7	MORADO
2	VERDE	4	7	MORADO
3	ROSADO	5	7	MORADO
6	AMARILLO	1	8	FUCSIA
4	AZUL	2	8	FUCSIA
5	ROJO	3	8	FUCSIA
2	VERDE	4	8	FUCSIA
3	ROSADO	5	8	FUCSIA
6	AMARILLO	1	9	SALMON
4	AZUL	2	9	SALMON
5	ROJO	3	9	SALMON
2	VERDE	4	9	SALMON
3	ROSADO	5	9	SALMON

UNION ALL

TABLA UNO		
ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5
1	NARANJA	6

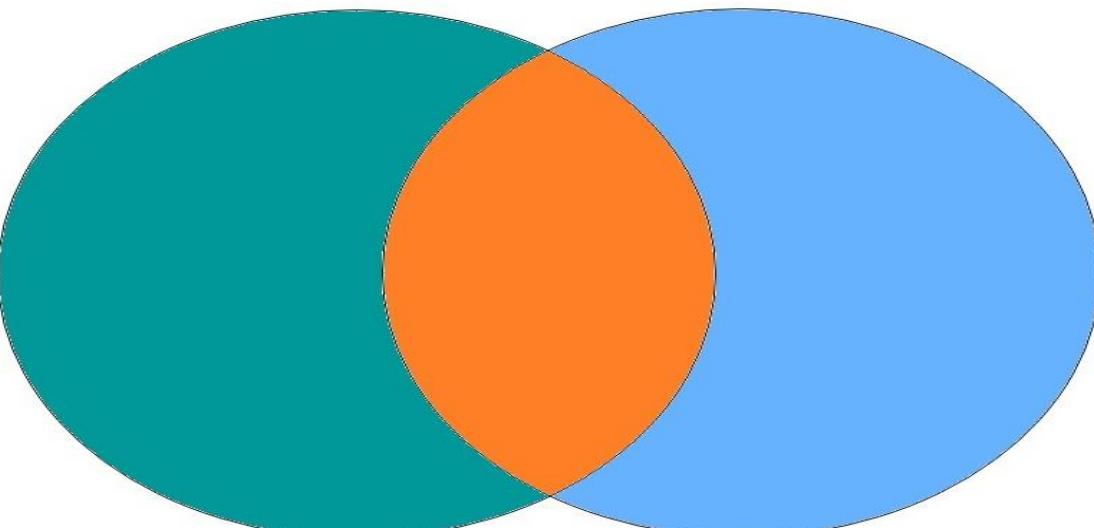
SELECT KEY, COLOR
FROM

TABLA_UNO
UNION ALL

SELECT KEY, COLOR
FROM

TABLA DOS	
KEY	COLOR
1	AMARILLO
3	ROJO
5	ROSADO
7	MORADO
8	FUCSIA
9	SALMON

UNION ALL	
KEY	COLOR
1	AMARILLO
2	AZUL
3	ROJO
4	VERDE
5	ROSADO
6	NARANJA
1	AMARILLO
3	ROJO
5	ROSADO
7	MORADO
8	FUCSIA
9	SALMON



UNION

TABLA UNO

ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5
1	NARANJA	6

SELECT KEY, COLOR

FROM

TABLA_UNO

UNION

SELECT KEY, COLOR

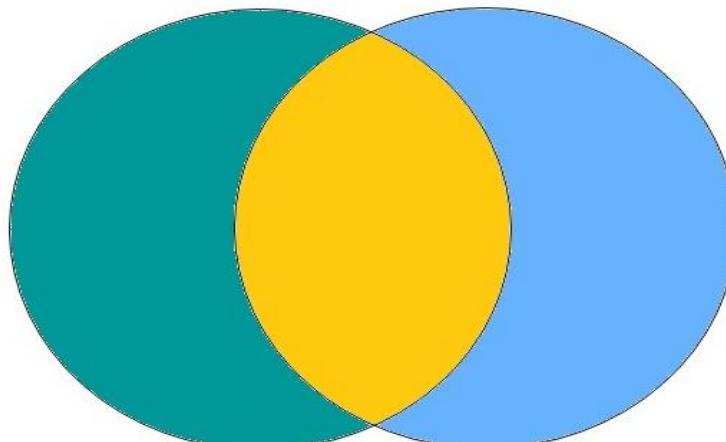
FROM

TABLA_DOS

ORDER BY KEY

TABLA DOS

KEY	COLOR
1	AMARILLO
3	ROJO
5	ROSADO
7	MORADO
8	FUCSIA
9	SALMON



UNION	
KEY	COLOR
1	AMARILLO
2	AZUL
3	ROJO
4	VERDE
5	ROSADO
6	NARANJA
7	MORADO
8	FUCSIA
9	SALMON

UNION INTERSECT

TABLA UNO

ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5
1	NARANJA	6

SELECT KEY, COLOR

FROM

TABLA_UNO

INTERSECT

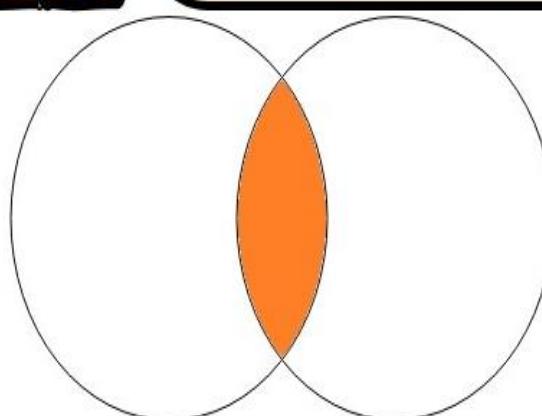
SELECT KEY, COLOR

FROM

TABLA_DOS

TABLA DOS

KEY	COLOR
1	AMARILLO
3	ROJO
5	ROSADO
7	MORADO
8	FUCSIA
9	SALMON



INTERSECT

KEY COLOR

1 AMARILLO

3 ROJO

5 ROSADO

UNION EXCEPT

TABLA UNO

ID	COLOR	KEY
6	AMARILLO	1
4	AZUL	2
5	ROJO	3
2	VERDE	4
3	ROSADO	5
1	NARANJA	6

SELECT KEY, COLOR

FROM

TABLA_UNO

EXCEPT

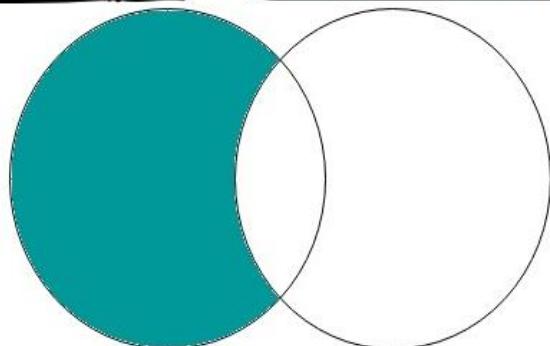
SELECT KEY, COLOR

FROM

TABLA_DOS

TABLA DOS

KEY	COLOR
1	AMARILLO
3	ROJO
5	ROSADO
7	MORADO
8	FUCSIA
9	SALMON



EXCEPT

KEY COLOR

2 AZUL

4 VERDE

6 NARANJA

CONTINENTES

REGION	ZONA	PAIS	TEMP
AFRICA	NORTE	EGIPTO	22
AMERICA	NORTE	MEXICO	37
EUROPA	ESTE	RUSIA	34
ASIA	SUR	INDIA	33
EUROPA	ESTE	GEORGIA	28
AFRICA	ESTE	KENYA	36
AMERICA	SUR	ARGENTINA	23
EUROPA	SUR	ITALIA	25
ASIA	ESTE	CHINA	17
AMERICA	NORTE	USA	21
EUROPA	OESTE	BELGICA	29
OCEANIA	SUR	AUSTRALIA	28
AMERICA	SUR	COLOMBIA	21
EUROPA	OESTE	Espana	30

SELECT

ZONA,

PAIS,

TEMP,

ROW_NUMBER () OVER (

PARTITION BY REGION

ORDER BY ZONA

) AS ROW,

REGION,

COUNT (TEMP) OVER (

PARTITION BY REGION

) AS COUNT

FROM CONTINENTES;

ROW - COUNT

ZONA	PAIS	TEMP	ROW	REGION	COUNT
ESTE	KENYA	36	1	AFRICA	2
NORTE	EGIPTO	22	2	AFRICA	2
NORTE	MEXICO	37	1	AMERICA	4
NORTE	USA	21	2	AMERICA	4
SUR	ARGENTINA	23	3	AMERICA	4
SUR	COLOMBIA	21	4	AMERICA	4
ESTE	CHINA	17	1	ASIA	2
SUR	INDIA	33	2	ASIA	2
ESTE	RUSIA	34	1	EUROPA	5
ESTE	GEORGIA	28	2	EUROPA	5
OESTE	Espana	30	3	EUROPA	5
OESTE	BELGICA	29	4	EUROPA	5
SUR	ITALIA	25	5	EUROPA	5
SUR	AUSTRALIA	28	1	OCEANIA	1

CONTINENTES

REGION	ZONA	PAIS	TEMP
AFRICA	NORTE	EGIPTO	22
AMERICA	NORTE	MEXICO	37
EUROPA	ESTE	RUSIA	34
ASIA	SUR	INDIA	33
EUROPA	ESTE	GEORGIA	28
AFRICA	ESTE	KENYA	36
AMERICA	SUR	ARGENTINA	23
EUROPA	SUR	ITALIA	25
ASIA	ESTE	CHINA	17
AMERICA	NORTE	USA	21
EUROPA	OESTE	BELGICA	29
OCEANIA	SUR	AUSTRALIA	28
AMERICA	SUR	COLOMBIA	21
EUROPA	OESTE	Espana	30

SELECT

PAIS,

TEMP,

RANK () OVER (

PARTITION BY REGION

ORDER BY ZONA

) AS RANK,

ZONA,

REGION,

DENSE_RANK () OVER (

PARTITION BY REGION

ORDER BY ZONA

) AS DENSE

FROM CONTINENTES;

RANK - DENSE RANK

PAIS	TEMP	RANK	ZONA	REGION	DENSE
KENYA	36	1	ESTE	AFRICA	1
EGIPTO	22	2	NORTE	AFRICA	2
MEXICO	37	1	NORTE	AMERICA	1
USA	21	1	NORTE	AMERICA	1
ARGENTINA	23	3	SUR	AMERICA	2
COLOMBIA	21	3	SUR	AMERICA	2
CHINA	17	1	ESTE	ASIA	1
INDIA	33	2	SUR	ASIA	2
RUSIA	34	1	ESTE	EUROPA	1
GEORGIA	28	1	ESTE	EUROPA	1
Espana	30	3	OESTE	EUROPA	2
BELGICA	29	3	OESTE	EUROPA	2
ITALIA	25	5	SUR	EUROPA	3
AUSTRALIA	28	1	SUR	OCEANIA	1

CONTINENTES

REGION	ZONA	PAIS	TEMP
AFRICA	NORTE	EGIPTO	22
AMERICA	NORTE	MEXICO	37
EUROPA	ESTE	RUSIA	34
ASIA	SUR	INDIA	33
EUROPA	ESTE	GEORGIA	28
AFRICA	ESTE	KENYA	36
AMERICA	SUR	ARGENTINA	23
EUROPA	SUR	ITALIA	25
ASIA	ESTE	CHINA	17
AMERICA	NORTE	USA	21
EUROPA	OESTE	BELGICA	29
OCEANIA	SUR	AUSTRALIA	28
AMERICA	SUR	COLOMBIA	21
EUROPA	OESTE	España	30

CONTINENTES

REGION	ZONA	PAIS	TEMP
AFRICA	NORTE	EGIPTO	22
AMERICA	NORTE	MEXICO	37
EUROPA	ESTE	RUSIA	34
ASIA	SUR	INDIA	33
EUROPA	ESTE	GEORGIA	28
AFRICA	ESTE	KENYA	36
AMERICA	SUR	ARGENTINA	23
EUROPA	SUR	ITALIA	25
ASIA	ESTE	CHINA	17
AMERICA	NORTE	USA	21
EUROPA	OESTE	BELGICA	29
OCEANIA	SUR	AUSTRALIA	28
AMERICA	SUR	COLOMBIA	21
EUROPA	OESTE	España	30

SELECT

```

SELECT
ZONA,
PAIS,
MIN (TEMP) OVER (
    PARTITION BY REGION
    ) AS MIN,
TEMP,
REGION,
MAX (TEMP) OVER (
    PARTITION BY REGION
    ) AS MAX
FROM CONTINENTES;
  
```

MINIMUN - MAXIMUN

ZONA	PAIS	MIN	TEMP	REGION	MAX
ESTE	KENYA	22	36	AFRICA	36
NORTE	EGIPTO	22	22	AFRICA	36
NORTE	MEXICO	21	37	AMERICA	37
NORTE	USA	21	21	AMERICA	37
SUR	ARGENTINA	21	23	AMERICA	37
SUR	COLOMBIA	21	21	AMERICA	37
ESTE	CHINA	17	17	ASIA	33
SUR	INDIA	17	33	ASIA	33
ESTE	RUSIA	25	34	EUROPA	34
ESTE	GEORGIA	25	28	EUROPA	34
OESTE	España	25	30	EUROPA	34
OESTE	BELGICA	25	29	EUROPA	34
SUR	ITALIA	25	25	EUROPA	34
SUR	AUSTRALIA	28	28	OCEANIA	28

CONTINENTES

REGION	ZONA	PAIS	TEMP
AFRICA	NORTE	EGIPTO	22
AMERICA	NORTE	MEXICO	37
EUROPA	ESTE	RUSIA	34
ASIA	SUR	INDIA	33
EUROPA	ESTE	GEORGIA	28
AFRICA	ESTE	KENYA	36
AMERICA	SUR	ARGENTINA	23
EUROPA	SUR	ITALIA	25
ASIA	ESTE	CHINA	17
AMERICA	NORTE	USA	21
EUROPA	OESTE	BELGICA	29
OCEANIA	SUR	AUSTRALIA	28
AMERICA	SUR	COLOMBIA	21
EUROPA	OESTE	España	30

SELECT

```

SELECT
ZONA,
PAIS,
SUM (TEMP) OVER (
    PARTITION BY REGION
    ) AS SUM,
TEMP,
REGION,
AVG (TEMP) OVER (
    PARTITION BY REGION
    ) AS AVG
FROM CONTINENTES;
  
```

SUM - AVERAGE

ZONA	PAIS	SUM	TEMP	REGION	AVG
ESTE	KENYA	58	36	AFRICA	29
NORTE	EGIPTO	58	22	AFRICA	29
NORTE	MEXICO	102	37	AMERICA	25.5
NORTE	USA	102	21	AMERICA	25.5
SUR	ARGENTINA	102	23	AMERICA	25.5
SUR	COLOMBIA	102	21	AMERICA	25.5
ESTE	CHINA	50	17	ASIA	25
SUR	INDIA	50	33	ASIA	25
ESTE	RUSIA	146	34	EUROPA	29.2
ESTE	GEORGIA	146	28	EUROPA	29.2
OESTE	España	146	30	EUROPA	29.2
OESTE	BELGICA	146	29	EUROPA	29.2
SUR	ITALIA	146	25	EUROPA	29.2
SUR	AUSTRALIA	28	28	OCEANIA	28

CONTINENTES

REGION	ZONA	PAIS	TEMP
AFRICA	NORTE	EGIPTO	22
AMERICA	NORTE	MEXICO	37
EUROPA	ESTE	RUSIA	34
ASIA	SUR	INDIA	33
EUROPA	ESTE	GEORGIA	28
AFRICA	ESTE	KENYA	36
AMERICA	SUR	ARGENTINA	23
EUROPA	SUR	ITALIA	25
ASIA	ESTE	CHINA	17
AMERICA	NORTE	USA	21
EUROPA	OESTE	BELGICA	29
OCEANIA	SUR	AUSTRALIA	28
AMERICA	SUR	COLOMBIA	21
EUROPA	OESTE	España	30

SELECT

```

SELECT
ZONA,
PAIS,
SUM (TEMP) OVER (
    PARTITION BY REGION
    ORDER BY ZONA ROWS BETWEEN
    1 PRECEDING AND 1 FOLLOWING
    ) AS SUM1,
TEMP,
REGION,
SUM (TEMP) OVER (
    PARTITION BY REGION
    ORDER BY ZONA ROWS BETWEEN
    2 PRECEDING AND 2 FOLLOWING
    ) AS SUM22
FROM CONTINENTES;
  
```

SUM11 - SUM22

ZONA	PAIS	SUM11	TEMP	REGION	SUM22
ESTE	KENYA	58	36	AFRICA	58
NORTE	EGIPTO	58	22	AFRICA	58
NORTE	MEXICO	58	37	AMERICA	81
NORTE	USA	81	21	AMERICA	102
SUR	ARGENTINA	65	23	AMERICA	102
SUR	COLOMBIA	44	21	AMERICA	65
ESTE	CHINA	50	17	ASIA	50
SUR	INDIA	50	33	ASIA	50
ESTE	RUSIA	62	34	EUROPA	92
ESTE	GEORGIA	92	28	EUROPA	121
OESTE	España	87	30	EUROPA	146
OESTE	BELGICA	84	29	EUROPA	112
SUR	ITALIA	54	25	EUROPA	84
SUR	AUSTRALIA	28	28	OCEANIA	28

CONTINENTES

REGION	ZONA	PAIS	TEMP
AFRICA	NORTE	EGIPTO	22
AMERICA	NORTE	MEXICO	37
EUROPA	ESTE	RUSIA	34
ASIA	SUR	INDIA	33
EUROPA	ESTE	GEORGIA	28
AFRICA	ESTE	KENYA	36
AMERICA	SUR	ARGENTINA	23
EUROPA	SUR	ITALIA	25
ASIA	ESTE	CHINA	17
AMERICA	NORTE	USA	21
EUROPA	OESTE	BELGICA	29
OCEANIA	SUR	AUSTRALIA	28
AMERICA	SUR	COLOMBIA	21
EUROPA	OESTE	España	30

SELECT

```

SELECT
ZONA,
PAIS,
SUM (TEMP) OVER (
    PARTITION BY REGION
    ORDER BY ZONA, ZONA
    ROWS BETWEEN UNBOUNDED
    PRECEDING AND CURRENT ROW
    ) AS ACUM,
TEMP,
REGION,
NTILE (2) OVER (
    PARTITION BY REGION
    ) AS NTILE
FROM CONTINENTES;
  
```

ACUM - NTILE

ZONA	PAIS	ACUM	TEMP	REGION	NTILE(2)
ESTE	KENYA	36	36	AFRICA	1
NORTE	EGIPTO	58	22	AFRICA	2
NORTE	MEXICO	37	37	AMERICA	1
NORTE	USA	58	21	AMERICA	1
SUR	ARGENTINA	81	23	AMERICA	2
SUR	COLOMBIA	102	21	AMERICA	2
ESTE	CHINA	17	17	ASIA	1
SUR	INDIA	50	33	ASIA	2
ESTE	RUSIA	34	34	EUROPA	1
ESTE	GEORGIA	62	28	EUROPA	1
OESTE	España	92	30	EUROPA	1
OESTE	BELGICA	121	29	EUROPA	2
SUR	ITALIA	146	25	EUROPA	2
SUR	AUSTRALIA	28	28	OCEANIA	1

CONTINENTES			
REGION	ZONA	PAIS	TEMP
AFRICA	NORTE	EGIPTO	22
AMERICA	NORTE	MEXICO	37
EUROPA	ESTE	RUSIA	34
ASIA	SUR	INDIA	33
EUROPA	ESTE	GEORGIA	28
AFRICA	ESTE	KENYA	36
AMERICA	SUR	ARGENTINA	23
EUROPA	SUR	ITALIA	25
ASIA	ESTE	CHINA	17
AMERICA	NORTE	USA	21
EUROPA	OESTE	BELGICA	29
OCEANIA	SUR	AUSTRALIA	28
AMERICA	SUR	COLOMBIA	21
EUROPA	OESTE	España	30

ZONA,
 TEMP,
 LAG (PAIS) OVER (PARTITION BY REGION
 ORDER BY ZONA
) AS LAG,
 PAIS,
 REGION,
 LEAD (PAIS) OVER (PARTITION BY REGION
 ORDER BY ZONA
) AS LEAD
 FROM CONTINENTES;

LAG - LEAD			
ZONA	TEMP	LAG	PAIS
ESTE	36	NULL	KENYA
NORTE	22	KENYA	EGIPTO
NORTE	37	NULL	MEXICO
NORTE	21	MEXICO	USA
SUR	23	USA	ARGENTINA
SUR	21	ARGENTINA	COLOMBIA
ESTE	17	NULL	CHINA
SUR	33	CHINA	INDIA
ESTE	34	NULL	RUSIA
ESTE	28	RUSIA	GEORGIA
OESTE	30	GEORGIA	España
OESTE	29	España	BELGICA
SUR	25	BELGICA	ITALIA
SUR	28	NULL	AUSTRALIA

CONTINENTES			
REGION	ZONA	PAIS	TEMP
AFRICA	NORTE	EGIPTO	22
AMERICA	NORTE	MEXICO	37
EUROPA	ESTE	RUSIA	34
ASIA	SUR	INDIA	33
EUROPA	ESTE	GEORGIA	28
AFRICA	ESTE	KENYA	36
AMERICA	SUR	ARGENTINA	23
EUROPA	SUR	ITALIA	25
ASIA	ESTE	CHINA	17
AMERICA	NORTE	USA	21
EUROPA	OESTE	BELGICA	29
OCEANIA	SUR	AUSTRALIA	28
AMERICA	SUR	COLOMBIA	21
EUROPA	OESTE	España	30

SELECT
 ZONA,
 TEMP,
 FIRST_VALUE (PAIS) OVER (PARTITION BY REGION
) AS FIRST,
 PAIS,
 REGION,
 LAST_VALUE (PAIS) OVER (PARTITION BY REGION
) AS LAST
 FROM CONTINENTES;

FIRST - LAST			
ZONA	TEMP	FIRST	PAIS
ESTE	36	KENYA	KENYA
NORTE	22	KENYA	EGIPTO
NORTE	37	MEXICO	MEXICO
NORTE	21	MEXICO	USA
SUR	23	MEXICO	ARGENTINA
SUR	21	MEXICO	COLOMBIA
ESTE	17	CHINA	CHINA
SUR	33	CHINA	INDIA
ESTE	34	RUSIA	RUSIA
ESTE	28	RUSIA	GEORGIA
OESTE	30	RUSIA	España
OESTE	29	RUSIA	BELGICA
SUR	25	RUSIA	ITALIA
SUR	28	AUSTRALIA	AUSTRALIA

CASE + BETWEEN + GROUP BY + HAVING + OFFSET + LIMIT + INDEX + SELECT INTO

SELECT OrderID, Quantity
 CASE
 WHEN Quantity > 30 THEN 'UP'
 WHEN Quantity < 30 THEN 'DOWN'
 ELSE 'EQUAL'
 END AS QuantityText
 FROM OrderDetails

SELECT Country, COUNT(CustomerID) AS TOTAL
 FROM Customer
 WHERE CountryID BETWEEN 1 AND 150
 GROUP BY Country
 HAVING COUNT(CustomerID) > 5
 ORDER BY COUNT(CustomerID) DESC
 OFFSET 5
 LIMIT 10

CREATE INDEX idx_colors
 ON TABLE_COLORS (COLOR);

CREATE OR REPLACE VIEW vw_Colores AS
 (SELECT ID, COLOR
 FROM COLORES
 WHERE ID IS NOT NULL
);

ALTER TABLE TABLE_COLORS
 ADD CHECK (KEY>=0 AND ID>=0);
 SELECT
 SupplierName, City, Country
 INTO Customer
 FROM Suppliers;

COMMON TABLE EXPRESSION + WINDOW FUNCTIONS + CASE + UNION

CONTINENTES

REGION	ZONA	PAIS	TEMP
AFRICA	NORTE	EGIPTO	22
AMERICA	NORTE	MEXICO	37
EUROPA	ESTE	RUSIA	34
ASIA	SUR	INDIA	33
EUROPA	ESTE	GEORGIA	28
AFRICA	ESTE	KENYA	36
AMERICA	SUR	ARGENTINA	23
EUROPA	SUR	ITALIA	25
ASIA	ESTE	CHINA	17
AMERICA	NORTE	USA	21
EUROPA	OESTE	BELGICA	29
OCEANIA	SUR	AUSTRALIA	28
AMERICA	SUR	COLOMBIA	21
EUROPA	OESTE	España	30

WITH MaxMinTemperatures AS (

SELECT

ZONA, PAIS, REGION, TEMP,

MAX(TEMP) OVER (

PARTITION BY REGION

) AS MAX_TEMP,

MIN(TEMP) OVER (

PARTITION BY REGION

) AS MIN_TEMP

FROM CONTINENTES

)

SELECT

ZONA, PAIS, REGION, TEMP,

CASE

WHEN TEMP = MAX_TEMP

THEN PAIS || ' MAX ' || en || REGION

END AS clasificacion

FROM MaxMinTemperatures

WHERE TEMP = MAX_TEMP

UNION

SELECT

ZONA, PAIS, REGION, TEMP,

CASE

WHEN TEMP = MIN_TEMP

THEN PAIS || ' MIN ' || en || REGION

END AS clasificacion

FROM MaxMinTemperatures

WHERE TEMP = MIN_TEMP

ORDER BY REGION, TEMP DESC ;

ANONYMOUS BLOCK

DECLARE

total_inserts INT := 0;

salario_final INT;

BEGIN

IF name IS NULL OR name = "" OR CHAR_LENGTH(name) < 4 THEN
RAISE EXCEPTION 'El nombre debe tener más de 3 caracteres';

END IF;

IF salary < 1000 THEN

RAISE EXCEPTION 'El salario no puede ser inferior a 1000';

ELSE

FOR iterador IN 1..insert_limit LOOP

salario_final := salary + iterador;

INSERT INTO test_empleado (nombre, salario) VALUES (name, salario_final);

total_inserts := total_inserts + 1;

RAISE NOTICE 'Valor de i: %, salario: %', iterador, salario_final;

END LOOP;

END IF;

END;

COMMIT;

ANONYMOUS BLOCK

DECLARE

```
total_inserts INT := 0;
salario_final INT;
name VARCHAR2(50) := 'John Doe';    salary NUMBER := 2000;
insert_limit NUMBER := 5;
BEGIN
IF name IS NULL OR name = "" OR LENGTH(name) < 4 THEN
    RAISE_APPLICATION_ERROR(-20001, 'nombre debe tener más de 3 caracteres');
END IF;
IF salary < 1000 THEN
    RAISE_APPLICATION_ERROR(-20001, 'El salario no puede ser inferior a 1000');
ELSE
    FOR iterador IN 1..insert_limit LOOP
        salario_final := salary + iterador;
        INSERT INTO test_empleado (nombre, salario) VALUES (name, salario_final);
        total_inserts := total_inserts + 1;
        DBMS_OUTPUT.PUT_LINE('Valor de i: ' || iterador || ', salario: ' || salario_final);
    END LOOP;
END IF;
END;
```

USER DEFINED FUNCTION

```
CREATE OR REPLACE FUNCTION insert_employee(name VARCHAR, salary NUMERIC, insert_limit NUMERIC)
RETURNS INT
AS $$

DECLARE
total_inserts INT := 0;
salario_final INT;
BEGIN
IF name IS NULL OR name = "" OR CHAR_LENGTH(name) < 4 THEN
    RAISE EXCEPTION 'El nombre debe tener más de 3 caracteres';
END IF;
IF salary < 1000 THEN
    RAISE EXCEPTION 'El salario no puede ser inferior a 1000';
ELSE
    FOR iterador IN 1..insert_limit LOOP
        salario_final := salary + iterador;
        INSERT INTO test_empleado (nombre, salario) VALUES (name, salario_final);
        total_inserts := total_inserts + 1;
        RAISE NOTICE 'Valor de i: %, salario: %', iterador, salario_final;
    END LOOP;
END IF;

RETURN total_inserts;
END;
$$ LANGUAGE plpgsql;

-- usar la funcion
SELECT insert_employee('EE333',1897, 3);
```

