

TRABAJO PRÁCTICO UNIDAD 2:

SENTENCIAS SQL NIVEL MEDIO

```
1) CREATE TABLE CATEGORIA(  
    codigo_cat CHAR(3) PRIMARY KEY NOT NULL,  
    descripcion VARCHAR(50) NOT NULL  
);  
  
CREATE TABLE BODEGA(  
    codigo_bodega CHAR(2) PRIMARY KEY NOT NULL,  
    descripcion VARCHAR(50) NOT NULL  
);  
  
CREATE TABLE VINO(  
    codigo_vino CHAR(2) PRIMARY KEY NOT NULL,  
    descripcion VARCHAR(50) NOT NULL,  
    ano_elabor DATE NOT NULL,  
    codigo_bodega CHAR(2) REFERENCES BODEGA(codigo_bodega),  
    codigo_cat CHAR(3) REFERENCES CATEGORIA(codigo_cat)  
);  
  
CREATE TABLE CATEGORIA_PROVEEDOR(  
    codigo_cat_prov CHAR(2) PRIMARY KEY NOT NULL,  
    cantidad INT NOT NULL,  
    facturacion NUMBER(8,2),  
    ventas INT);
```

```
CREATE TABLE PROVEEDOR(
```

```
    codigo_prov CHAR(2) PRIMARY KEY NOT NULL,
```

```
    razon_social VARCHAR(30) NOT NULL,
```

```
    codigo_cat_prov CHAR(2) REFERENCES  
    CATEGORIA_PROVEEDOR(codigo_cat_prov)
```

```
);
```

```
CREATE TABLE COMPRA(
```

```
    codigo_prov CHAR(2) REFERENCES PROVEEDOR(codigo_prov),
```

```
    codigo_vino CHAR(2) REFERENCES VINO(codigo_vino),
```

```
    fecha DATE NOT NULL,
```

```
    cantidad INT NOT NULL
```

```
);
```

```
CREATE TABLE CLIENTE(
```

```
    num_corr INT PRIMARY KEY NOT NULL,
```

```
    nom_razon VARCHAR(30) NOT NULL,
```

```
    telefono VARCHAR(25) NOT NULL,
```

```
    email VARCHAR(30) NOT NULL,
```

```
    tarjeta VARCHAR(40) NOT NULL
```

```
);
```

```
CREATE TABLE PEDIDO(  
    codigo_vino CHAR(2) REFERENCES VINO(codigo_vino),  
    codigo_cliente INT REFERENCES CLIENTE(num_corr),  
    fecha DATE NOT NULL,  
    cantidad INT NOT NULL,  
    precio_unit NUMBER(8,2) NOT NULL  
);
```

```
2) INSERT INTO CATEGORIA VALUES ('TIN','TINTOS');
```

```
INSERT INTO CATEGORIA VALUES ('BLA','BLANCOS');
```

```
INSERT INTO CATEGORIA VALUES ('ROS','ROSADOS');
```

```
INSERT INTO CATEGORIA VALUES ('ESP','ESPUMANTES');
```

```
INSERT INTO BODEGA VALUES ('AL','ALAMOS');
```

```
INSERT INTO BODEGA VALUES ('AF','ACHAVAL FERRER');
```

```
INSERT INTO BODEGA VALUES ('BA','BODEGA AMALAYA');
```

```
INSERT INTO BODEGA VALUES ('BR','BODEGA ARGENTO');
```

```
INSERT INTO VINO VALUES ('LM','LINDAFLORES MALBEC', 2000, 'TIN', 'AL');
```

```
INSERT INTO VINO VALUES ('MM','MARTIN BRUNO MALBEC', 1998, 'TIN', 'AF');
```

```
INSERT INTO VINO VALUES ('VB','VIA BLANCA MALBEC', 1996, 'TIN', 'BA');
```

```
INSERT INTO VINO VALUES ('GL','GRETA LA SALVAJE SEMILLON', 1998, 'BLA', 'BA');
```

INSERT INTO VINO VALUES ('TR','TRUMPETER RESERVE DOUX', 1996, 'BLA', 'BA');

INSERT INTO VINO VALUES ('AV','ALTA VISTA SAUVIGNON', 1995, 'BLA', 'AF');

INSERT INTO VINO VALUES ('AS','AMANSADO CLASSIC ROSE', 2002, 'BLA', 'AF');

INSERT INTO VINO VALUES ('ZT','ZORZAL TERROIR', 2004, 'BLA', 'AF');

INSERT INTO VINO VALUES ('PF','PINK FLOOD DE JUANCHI', 2006, 'BLA', 'AF');

INSERT INTO CLIENTE VALUES (1, 'MARINAO GECIK', '15-4-567-8976',
'MGECIK@GMAIL.COM', '3456-5678-3245-7654');

INSERT INTO CLIENTE VALUES (2, 'RICARDO PEREZ', '15-4-356-8900',
'RPEREZ@GMAIL.COM', '1234-5678-6789-2345');

INSERT INTO CLIENTE VALUES (3, 'SUSANA VAINSHTEIN', '15-4-765-1234',
'SVAINSHTEIN@GMAIL.COM', '9876-5092-3456-1342');

INSERT INTO CLIENTE VALUES (4, 'PABLO PATRIZI', '15-4-894-1334',
'PPATRIZI@GMAIL.COM', '9986-5077-3896-1652');

INSERT INTO PEDIDO VALUES('LM', '1', TO_DATE('21/06/2018','DD/MM/YYYY'), 100,
150);

INSERT INTO PEDIDO VALUES('MM', '1', TO_DATE('21/06/2018','DD/MM/YYYY'), 100,
200);

INSERT INTO PEDIDO VALUES('AV', '2', TO_DATE('21/05/2018','DD/MM/YYYY'), 130, 250);

INSERT INTO PEDIDO VALUES('AS', '2', TO_DATE('21/05/2018','DD/MM/YYYY'), 120, 300);

INSERT INTO PEDIDO VALUES('GL', '3', TO_DATE('21/03/2019','DD/MM/YYYY'), 140, 100);

INSERT INTO PEDIDO VALUES('TR', '3', TO_DATE('21/03/2019','DD/MM/YYYY'), 120, 120);

INSERT INTO CATEGORIA_PROVEEDOR VALUES ('AC', 'ALTA CALIDAD', 100, 1000000,
30000);

INSERT INTO CATEGORIA_PROVEEDOR VALUES ('BA', 'BAJA CALIDAD', 200,0,0);

```

INSERT INTO PROVEEDOR VALUES ( 'LR', 'LA RURAL', 'AC');

INSERT INTO PROVEEDOR VALUES ( 'CO', 'COYANCO', 'AC');

INSERT INTO PROVEEDOR VALUES ( 'MO', 'MONETVIEJO', 'BA');

INSERT INTO PROVEEDOR VALUES ( 'RO', 'ROLLAND', 'BA');

INSERT INTO COMPRA VALUES( 'LR', 'LM', TO_DATE('21/06/2017','DD/MM/YYYY'), 2000);

INSERT INTO COMPRA VALUES( 'LR', 'MM', TO_DATE('21/06/2017','DD/MM/YYYY'), 2000);

INSERT INTO COMPRA VALUES( 'MO', 'GL', TO_DATE('21/09/2017','DD/MM/YYYY'), 1500);

INSERT INTO COMPRA VALUES( 'MO', 'TR', TO_DATE('21/09/2017','DD/MM/YYYY'), 1500);

INSERT INTO COMPRA VALUES( 'RO', 'GL', TO_DATE('21/05/2017','DD/MM/YYYY'), 1000);

INSERT INTO COMPRA VALUES( 'RO', 'TR', TO_DATE('21/05/2017','DD/MM/YYYY'), 800);

```

3) SELECT * FROM CLIENTE;

NUM_CORR	NOM_RAZON	TELEFONO	EMAIL	TARJETA
2	RICARDO PEREZ	15-4-356-8900	RPEREZ@GMAIL.COM	1234-5678-9879-2345
3	SUSANA VAINSHTEIN	15-4-785-1234	SVAINSHTEIN@GMAIL.COM	9876-5092-3456-1342
1	MARINAO GECIK	15-4-567-8976	MGECKI@GMAIL.COM	3456-5678-3245-2654
4	PABLO PATRIZI	15-4-894-1354	PPATRIZI@GMAIL.COM	9986-5077-3896-1652

4) SELECT fecha, cantidad FROM PEDIDO;

FECHA	CANTIDAD
04/21/2018	100
04/21/2018	100
04/21/2019	120
05/21/2018	130

5) SELECT (cantidad*precio_unit) AS "TOTAL" FROM PEDIDO;

TOTAL
15000
20000
14400
12500

6) SELECT cantidad*((precio_unit*35)/100) FROM PEDIDO;

Results	Explain	Describe	Save SQL	History
CANTIDAD*((PRECIO_UNIT*35)/100)				
3250				
7000				
5040				
11575				

7) SELECT nom_razon AS “Nombre o Razón social” FROM CLIENTE;

Results	Explain	Describe	Save SQL	History
Nombre o Razón social				
RICARDO PEREZ				
SUSANA VAINSHTEIN				
MARINAD GECK				
PABLO PATRIZI				

8) SELECT nom_razon || '-' || telefono FROM CLIENTE;

Results	Explain	Describe	Save SQL	History
NOM_RAZON '-' TELEFONO				
RICARDO PEREZ - 15-4-356-8900				
SUSANA VAINSHTEIN - 15-4-765-1234				
MARINAD GECK - 15-4-567-8976				
PABLO PATRIZI - 15-4-894-1334				

9) SELECT DISTINCT fecha FROM PEDIDO;

Results	Explain	Describe	Save SQL	History
FECHA				
05/21/2018				
05/21/2018				
05/21/2019				

10) SELECT * FROM CATEGORIA_PROVEEDOR

WHERE facturacion > 100;

Results	Explain	Describe	Save SQL	History
CODIGO_CAT_PROV				
AC				

11) SELECT * FROM CLIENTE

WHERE nom_razon = 'SUSANA VAINSHTEIN ';

Results	Explain	Describe	Save SQL	History
NUM_CORR				
1				

12) SELECT * FROM CLIENTE

WHERE num_corr BETWEEN 1 AND 4

NUM_CORR	NOM_RAZON	TELEFONO	EMAIL	TARJETA
2	RICARDO PEREZ	15-4-356-8900	RPEREZ@GMAIL.COM	1234-5678-6789-2345
3	SUSANA VAINSHTEIN	15-4-765-1234	SVAINSHTEIN@GMAIL.COM	9876-5092-3456-1542
1	MARINAO GECK	15-4-567-8976	MGECK@GMAIL.COM	3456-5678-3245-7654
4	PABLO PATRIZI	15-4-894-1334	PPATRIZI@GMAIL.COM	9986-5077-3896-1652

13) SELECT * FROM CLIENTE

WHERE nom_razon IN ('RICARDO PEREZ', 'PABLO PATRIZI')

NUM_CORR	NOM_RAZON	TELEFONO	EMAIL	TARJETA
2	RICARDO PEREZ	15-4-356-8900	RPEREZ@GMAIL.COM	1234-5678-6789-2345
4	PABLO PATRIZI	15-4-894-1334	PPATRIZI@GMAIL.COM	9986-5077-3896-1652

2 rows returned in 0.01 seconds

SELECT * FROM CLIENTE

WHERE nom_razon NOT IN ('RICARDO PEREZ', 'PABLO PATRIZI')

NUM_CORR	NOM_RAZON	TELEFONO	EMAIL	TARJETA
3	SUSANA VAINSHTEIN	15-4-765-1234	SVAINSHTEIN@GMAIL.COM	9876-5092-3456-1542
1	MARINAO GECK	15-4-567-8976	MGECK@GMAIL.COM	3456-5678-3245-7654

2 rows returned in 0.01 seconds

14) SELECT * FROM PROVEEDOR

WHERE razon_social LIKE 'LA%'

CODIGO_PROV	RAZON_SOCIAL	CODIGO_CAT_PROV
LR	LA RURAL	AC

1 rows returned in 0.01 seconds

15) SELECT * FROM CATEGORIA_PROVEEDOR

WHERE ventas IS NULL

no data found

SELECT * FROM CATEGORIA_PROVEEDOR

WHERE ventas IS NOT NULL

CODIGO_CAT_PROV	CANTIDAD	FACTURACION	VENTAS	CALIDAD
AC	100	100000	30000	ALTA CALIDAD
BA	200	0	0	BAJA CALIDAD

2 rows returned in 0.00 seconds [Download](#)

16) SELECT * FROM VINO

ORDER BY ano_elabor

CODIGO_VINO	DESCRIPCION	ANO_ELAVOR	CODIGO_BODEGA	CODIGO_CAT
AV	ALTA VISTA SAUVIGNON	1995	AF	BLA
TR	TRUMPETER RESERVE DOLUX	1996	BA	BLA
VB	VIA BLANCA MALBEC	1996	BA	TIN
MM	MARTIN BRUNO MALBEC	1998	AF	TIN

SELECT * FROM VINO

ORDER BY ano_elabor DESC

CODIGO_VINO	DESCRIPCION	ANO_ELAVOR	CODIGO_BODEGA	CODIGO_CAT
PF	PINK FLOOD DE JUANCHI	2006	AF	BLA
ET	ZORZAL TERROR	2004	AF	BLA
AS	AMANSADO CLASSIC ROSE	2002	AF	BLA
LM	LINDAFLOL MALBEC	2000	AL	TIN

17) SELECT * FROM VINO

WHERE codigo_bodega = :codigo_bodega

:CODIGO_BODEGA	AF
----------------	----

CODIGO_VINO	DESCRIPCION	ANO_ELAVOR	CODIGO_BODEGA	CODIGO_CAT
MM	MARTIN BRUNO MALBEC	1998	AF	TIN
AS	AMANSADO CLASSIC ROSE	2002	AF	BLA
AV	ALTA VISTA SAUVIGNON	1995	AF	BLA
PF	PINK FLOOD DE JUANCHI	2006	AF	BLA

18) SELECT cantidad, NVL(facturacion, 100) FROM

CATEGORIA_PROVEEDOR



The screenshot shows the results of a SQL query using the NVL function. The table has two columns: 'CANTIDAD' and 'NVL(FACTURACION,100)'. There are two rows of data. The first row shows 'CANTIDAD' as 100 and 'NVL(FACTURACION,100)' as 100000. The second row shows 'CANTIDAD' as 200 and 'NVL(FACTURACION,100)' as 0. The interface includes tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. At the bottom, it indicates '2 rows returned in 0.01 seconds' and provides a 'Download' link.

CANTIDAD	NVL(FACTURACION,100)
100	100000
200	0

(DEVUELVE LO MISMO YA QUE NO EXISTEN VALORES NULL EN FACTURACION)

19) SELECT cantidad, NVL2(facturacion, 'TIENE FACTURACION', 'NO TIENE') FROM

CATEGORIA_PROVEEDOR



The screenshot shows the results of a SQL query using the NVL2 function. The table has two columns: 'CANTIDAD' and 'NVL2(FACTURACION,'TIENE FACTURACION','NO TIENE')'. There are two rows of data. The first row shows 'CANTIDAD' as 100 and 'NVL2(FACTURACION,'TIENE FACTURACION','NO TIENE')' as 'TIENE FACTURACION'. The second row shows 'CANTIDAD' as 200 and 'NVL2(FACTURACION,'TIENE FACTURACION','NO TIENE')' as 'TIENE FACTURACION'. The interface includes tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. At the bottom, it indicates '2 rows returned in 0.01 seconds' and provides a 'Download' link.

CANTIDAD	NVL2(FACTURACION,'TIENE FACTURACION','NO TIENE')
100	TIENE FACTURACION
200	TIENE FACTURACION

20) SELECT codigo_vino, codigo_cliente, fecha, NULLIF(10,10) FROM

PEDIDO



The screenshot shows the results of a SQL query using the NULLIF function. The table has four columns: 'CODIGO_VINO', 'CODIGO_CLIENTE', 'FECHA', and 'NULLIF(10,10)'. There are four rows of data. The first row shows 'CODIGO_VINO' as 'LM', 'CODIGO_CLIENTE' as 1, 'FECHA' as 06/21/2018, and 'NULLIF(10,10)' as '-'. The second row shows 'CODIGO_VINO' as 'MM', 'CODIGO_CLIENTE' as 1, 'FECHA' as 06/21/2018, and 'NULLIF(10,10)' as '-'. The third row shows 'CODIGO_VINO' as 'TR', 'CODIGO_CLIENTE' as 3, 'FECHA' as 05/21/2019, and 'NULLIF(10,10)' as '-'. The fourth row shows 'CODIGO_VINO' as 'AV', 'CODIGO_CLIENTE' as 2, 'FECHA' as 05/21/2018, and 'NULLIF(10,10)' as '-'. The interface includes tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. At the bottom, it indicates '4 rows returned in 0.01 seconds' and provides a 'Download' link.

CODIGO_VINO	CODIGO_CLIENTE	FECHA	NULLIF(10,10)
LM	1	06/21/2018	-
MM	1	06/21/2018	-
TR	3	05/21/2019	-
AV	2	05/21/2018	-

(DEVUELVE UNA COLUMNA CON TODOS VALORES NULOS YA QUE LOS ARGUMENTOS DE LA SENTENCIA NULLIF() SON IGUALES)

21) SELECT COALESCE(NULL, NULL, NULL, 3, 4, NULL) FROM

PEDIDO



The screenshot shows the results of a SQL query using the COALESCE function. The table has one column: 'COALESCE(NULL,NULL,NULL,3,4,NULL)'. There are four rows of data, all showing the value 3. The interface includes tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. At the bottom, it indicates '4 rows returned in 0.01 seconds' and provides a 'Download' link.

COALESCE(NULL,NULL,NULL,3,4,NULL)
3
3
3
3

```

22) SELECT cantidad, precio_unit,

CASE WHEN precio_unit < 120 THEN 'ECONOMICO'

WHEN precio_unit BETWEEN 120 AND 200 THEN 'ESTANDAR'

ELSE 'CARO'

END TIPO_DE_PRODUCTO

FROM PEDIDO

ORDER BY precio_unit

```

Results	Explain	Describe	Saved SQL	History
CANTIDAD	PRECIO_UNIT	TIPO_DE_PRODUCTO		
140	100	ECONOMICO		
120	120	ESTANDAR		
100	150	ESTANDAR		
100	200	ESTANDAR		
130	250	CARO		

```

23) SELECT codigo_prov, DECODE(codigo_prov, 'RO', 'ROLLAND', 'LR', 'LA RURAL', 'CO',
'COYANCO', 'MONETVIEJO') FROM PROVEEDOR

```

Results	Explain	Describe	Saved SQL	History
CODIGO_PROV	DECODE(CODIGO_PROV,'RO','ROLLAND','LR','LA RURAL','CO','COYANCO','MONETVIEJO')			
CO	COYANCO			
LR	LA RURAL			
MO	MONETVIEJO			
RO	ROLLAND			

```

24) SELECT num_corr, LOWER(nom_razon) AS "Nombre o razon en MINUSCULA", telefono

FROM CLIENTE

```

Results	Explain	Describe	Saved SQL	History
NUM_CORR	Nombre o razon en MINUSCULA	TELEFONO		
2	ricardo perez	15-4-356-8900		
3	susana vainshtein	15-4-765-1234		
1	marino gecik	15-4-567-8976		
4	pablo patrizi	15-4-894-1334		

```

25) SELECT num_corr, UPPER(nom_razon) AS "Nombre o razon en MAYUSCULA", telefono

FROM CLIENTE

```

Results	Explain	Describe	Saved SQL	History
NUM_CORR	Nombre o razon en MAYUSCULA	TELEFONO		
2	RICARDO PEREZ	15-4-356-8900		
3	SUSANA VAINSHTEIN	15-4-765-1234		
1	MARINAO GEKIC	15-4-567-8976		
4	PABLO PATRIZI	15-4-894-1334		

26) SELECT num_corr, INITCAP(nom_razon) AS "Primera letra MAY", telefono FROM CLIENTE

NUM_CORR	Primera letra MAY	TELEFONO
2	Ricardo Perez	15-4-356-8900
5	Susana Vainshtein	15-4-765-1254
3	Martina Geck	15-4-567-8976
4	Pablo Patrizi	15-4-894-1354

27) SELECT CONCAT(nom_razon,email) AS "Nombre y EMAIL" FROM CLIENTE

Nombre y EMAIL
RICARDO PEREZPEREZ@GMAIL.COM
SUSANA VAINSHTEINVAINSHTEIN@GMAIL.COM
MARTINA GECKGECK@GMAIL.COM
PABLO PATRIZIPATRIZI@GMAIL.COM

28) SELECT SUBSTR(nom_razon, 1, 3) FROM CLIENTE

SUBSTR(NOM_RAZON,1,3)
RIC
SUS
MAR
PAB

29) SELECT LENGTH(descripcion) FROM VINO

LENGTH(DESCRIPCION)
19
21
20
22

30) SELECT descripcion, INSTR(descripcion, 'MA') FROM VINO

DESCRIPCION	INSTR(DESCRIPCION,'MA')
MARTIN BRUNO MALBEC	1
AMANSADO CLASSIC ROSE	2
ALTA VISTA SAUVIGNON	0
TRUMPETER RESERVE DOLUX	0

31) SELECT descripcion, LPAD(descripcion, 30, '0') FROM VINO

DESCRIPCION	LPAD(DESCRIPCION,30,'0')
MARTIN BRUNO MALBEC	0000000000MARTIN BRUNO MALBEC
AMANSADO CLASSIC ROSE	0000000000AMANSADO CLASSIC ROSE
ALTA VISTA SAUVIGNON	0000000000ALTA VISTA SAUVIGNON
TRUMPETER RESERVE DOLUX	0000000000TRUMPETER RESERVE DOLUX

32) SELECT TRIM(email) FROM CLIENTE



TRIM(EMAIL)
RPerez@gmail.com
SVANGHTEIN@gmail.com
MGECK@gmail.com
PATRIZI@gmail.com

SELECT LTRIM(email) FROM CLIENTE



LTRIM(EMAIL)
RPerez@gmail.com
SVANGHTEIN@gmail.com
MGECK@gmail.com
PATRIZI@gmail.com

SELECT RTRIM(email) FROM CLIENTE



RTRIM(EMAIL)
RPerez@gmail.com
SVANGHTEIN@gmail.com
MGECK@gmail.com
PATRIZI@gmail.com

33) SELECT nom_razon, REPLACE(nom_razon, 'RICARDO', 'RICKY') FROM CLIENTE



NOM_RAZON	REPLACE(NOM_RAZON,'RICARDO','RICKY')
RICHARDO PEREZ	RICKY PEREZ
SUSANA VANGHTEIN	SUSANA VANGHTEIN
MARINAO GECK	MARINAO GECK
PABLO PATRIZI	PABLO PATRIZI

34) SELECT ROUND(100000.557070, 2) FROM PEDIDO



ROUND(100000.557070,2)
100000.56
100000.56
100000.56
100000.56

35) SELECT TRUNC(100000.557070, 2) FROM PEDIDO



TRUNC(100000.557070,2)
100000.55
100000.55
100000.55
100000.55

36) SELECT MOD(15, 4) FROM PEDIDO

Results	Explain	Describe	Saved SQL	History
MOD(15,4)				
5				
5				
5				
5				

37) SELECT TRUNC(MONTHS_BETWEEN(SYSDATE, fecha)) FROM COMPRA

Results	Explain	Describe	Saved SQL	History
TRUNC(MONTHS_BETWEEN(SYSDATE,FECHA))				
70				
69				
69				
66				

38) SELECT fecha, ADD_MONTHS(fecha, 2) AS "Fecha con dos meses más" FROM COMPRA

Results	Explain	Describe	Saved SQL	History
FECHA				
05/21/2017				
06/21/2017				
08/21/2017				
09/21/2017				

39) SELECT fecha, NEXT_DAY(fecha, 'THU') AS "PROXIMO JUEVES", LAST_DAY(fecha) AS "ÚLTIMO DIA DEL MES" FROM COMPRA

Results	Explain	Describe	Saved SQL	History
FECHA				
05/21/2017				
06/21/2017				
08/21/2017				
09/21/2017				

40) SELECT TO_DATE('21/06/2018','DD/MM/YYYY') FROM PEDIDO

Results	Explain	Describe	Saved SQL	History
TO_DATE('21/06/2018','DD/MM/YYYY')				
06/21/2018				
06/21/2018				
06/21/2018				
06/21/2018				

41) SELECT codigo_cliente, TO_CHAR(fecha, 'DAY') AS Dia FROM PEDIDO

Results	Explain	Describe	Saved SQL	History
CODIGO_CLIENTE				
1				
1				
5				
2				

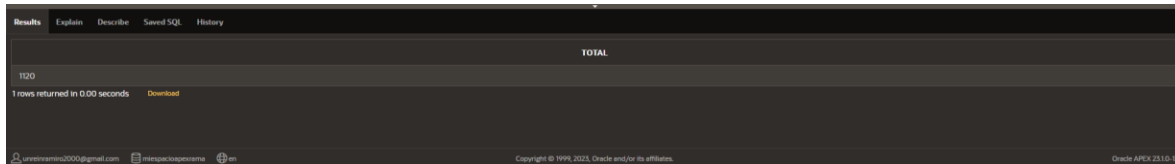
42) SELECT TO_NUMBER('123.456',999.999) FROM CLIENTE



The screenshot shows the Oracle APEX interface with the query results for 'TO_NUMBER('123.456',999.999)'. The results are displayed in a table with one column and four rows, all containing the value '123.456'. The interface includes tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. At the bottom, it shows the user 'univresma2000@gmail.com', the workspace 'mispacioapexrema', and the Oracle APEX version '23.1.0-19'.

TO_NUMBER('123.456',999.999)
123.456
123.456
123.456
123.456

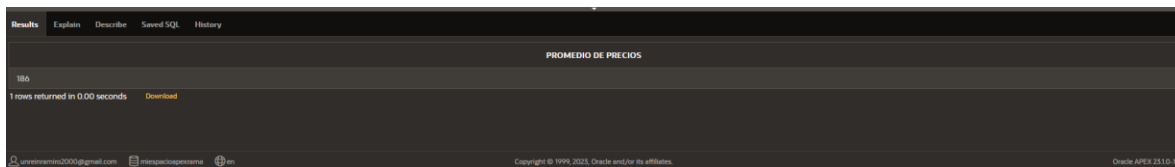
43) SELECT SUM(precio_unit) AS TOTAL FROM PEDIDO



The screenshot shows the Oracle APEX interface with the query results for 'SUM(precio_unit) AS TOTAL'. The results are displayed in a table with one column and one row, containing the value '1120'. The interface includes tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. At the bottom, it shows the user 'univresma2000@gmail.com', the workspace 'mispacioapexrema', and the Oracle APEX version '23.1.0-19'.

TOTAL
1120

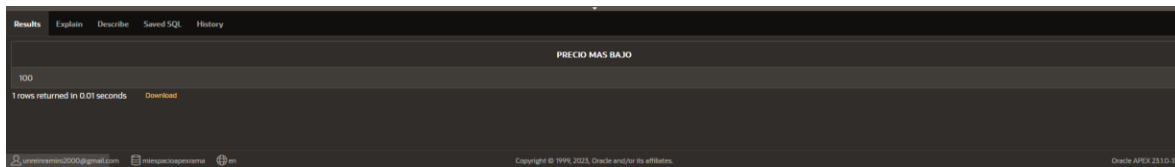
44) SELECT TRUNC(AVG(precio_unit)) AS "PROMEDIO DE PRECIOS" FROM PEDIDO



The screenshot shows the Oracle APEX interface with the query results for 'TRUNC(AVG(precio_unit)) AS "PROMEDIO DE PRECIOS"'. The results are displayed in a table with one column and one row, containing the value '186'. The interface includes tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. At the bottom, it shows the user 'univresma2000@gmail.com', the workspace 'mispacioapexrema', and the Oracle APEX version '23.1.0-19'.

PROMEDIO DE PRECIOS
186

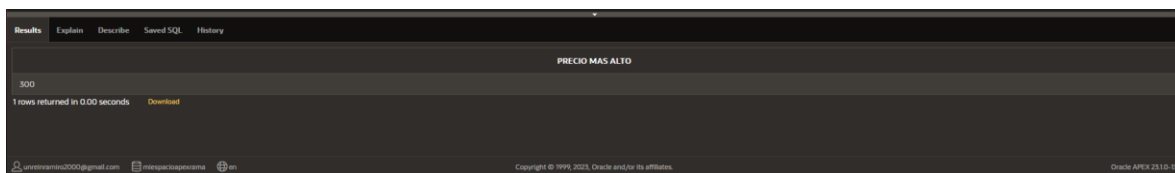
45) SELECT MIN(precio_unit) AS "PRECIO MAS BAJO" FROM PEDIDO



The screenshot shows the Oracle APEX interface with the query results for 'MIN(precio_unit) AS "PRECIO MAS BAJO"'. The results are displayed in a table with one column and one row, containing the value '100'. The interface includes tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. At the bottom, it shows the user 'univresma2000@gmail.com', the workspace 'mispacioapexrema', and the Oracle APEX version '23.1.0-19'.

PRECIO MAS BAJO
100

SELECT MAX(precio_unit) AS "PRECIO MAS ALTO" FROM PEDIDO



The screenshot shows the Oracle APEX interface with the query results for 'MAX(precio_unit) AS "PRECIO MAS ALTO"'. The results are displayed in a table with one column and one row, containing the value '300'. The interface includes tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. At the bottom, it shows the user 'univresma2000@gmail.com', the workspace 'mispacioapexrema', and the Oracle APEX version '23.1.0-19'.

PRECIO MAS ALTO
300

46) SELECT COUNT(codigo_cliente) AS "CANTIDAD DE CLIENTES" FROM PEDIDO



The screenshot shows the Oracle APEX interface with the query results for 'COUNT(codigo_cliente) AS "CANTIDAD DE CLIENTES"'. The results are displayed in a table with one column and one row, containing the value '6'. The interface includes tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. At the bottom, it shows the user 'univresma2000@gmail.com', the workspace 'mispacioapexrema', and the Oracle APEX version '23.1.0-19'.

CANTIDAD DE CLIENTES
6

Results	Explain	Describe	Save SQL	History
---------	---------	----------	----------	---------

CANTIDAD DE CLIENTES SIN REPETIR

5

1 rows returned in 0.00 seconds [Download](#)

Results	Explain	Describe	Saved SQL	History
CODIGO_CLIENTE			TOTAL	
1			35000	
5			28400	
2 rows returned in 0.01 seconds Download				

Results	Explain	Describe	Save as SQL	History
CODIGO_VINO	CANTIDAD	PRECIO_UNIT	DESCRIPCION	
MM	100	200	MARTIN BRUNO MALBEC	
AS	120	300	AMANEZADO CLASSAC ROSE	
AV	130	250	ALTA VISTA SAUVIGNON	
TR	120	120	TRUMPETER RESERVE DOLUX	

Results	Explain	Describe	Saved SQL	History		
CODIGO_VINO	CODIGO_CLIENTE	PRECIO_UNIT	CANTIDAD	NOM_BAZON	TELEFONO	TARJETA
AV	2	250	130	RICARDO PEREZ	15-4-356-8900	1254-5678-6789-2345
AS	2	300	120	RICARDO PEREZ	15-4-356-8900	1254-5678-6789-2345
TR	3	100	110	SUSANA VARGHSEEN	15-4-765-1234	9876-5092-3456-1342
GL	3	100	140	SUSANA VARGHSEEN	15-4-765-1234	9876-5092-3456-1342

 vinomirco2000@gmail.com

 interspacepersonal

 1m

Copyright © 1999, 2023, Oracle and/or its affiliates.

Oracle APEX 23.1.0

51) SELECT prov.razon_social, com.codigo_vino, com.fecha, com.cantidad FROM
PROVEEDOR prov

RIGHT OUTER JOIN COMPRA com ON prov.codigo_prov = com.codigo_prov

RAZON_SOCIAL	CODIGO_VINO	FECHA	CANTIDAD
ROLLAND	TR	06/21/2017	800
ROLLAND	GL	06/21/2017	1000
LA RURAL	LM	06/21/2017	2000
LA RURAL	MM	06/21/2017	2000

52) INSERT INTO VINO(codigo_vino, descripcion, ano_elabor, codigo_bodega, codigo_cat)
VALUES('NA', 'BODEGA MENDOZA', 2007, 'BA', 'BLA');

TR	TRUMPETER RESERVE DOLUX	1996	BA	BLA
PF	PINK FLOOD DE JUANCHI	2006	AF	BLA
VB	VIA BLANCA MALBEC	1996	BA	TIN
NA	BODEGA MENDOZA	2007	BA	BLA
LM	LINDAFLOI MALBEC	2000	AL	TIN
GL	GRETA LA SALVAJE SEMILLON	1998	BA	BLA

53) DELETE VINO

WHERE codigo_vino = 'NA'

AV	ALTA VISTA SALVIGNON	1995	AF	BLA
TR	TRUMPETER RESERVE DOLUX	1996	BA	BLA
PF	PINK FLOOD DE JUANCHI	2006	AF	BLA
VB	VIA BLANCA MALBEC	1996	BA	TIN
LM	LINDAFLOI MALBEC	2000	AL	TIN
GL	GRETA LA SALVAJE SEMILLON	1998	BA	BLA

54) UPDATE VINO

SET ano_elabor = 2000

WHERE codigo_vino = 'MM'

CODIGO_VINO	DESCRIPCION	ANO_ELABOR	CODIGO_BODEGA	CODIGO_CAT
MM	MARTIN BRUNO MALBEC	2000	AF	TIN
AS	AMANSADO CLASSIC ROSE	2002	AF	BLA
AV	ALTA VISTA SALVIGNON	1995	AF	BLA
TR	TRUMPETER RESERVE DOLUX	1996	BA	BLA