
TRABAJO PRÁCTICO 03

Fundamento de bases de datos: Unidad 03

Alumno

González, Matías Ezequiel

Contents

1	Lenguaje DML	3
1.0.1	Ejercicio 1	3
1.0.2	Ejercicio 2	3
1.0.3	Ejercicio 3	4
1.0.4	Ejercicio 4	4
1.0.5	Ejercicio 5	5
1.0.6	Ejercicio 6	5
1.0.7	Ejercicio 7	6
1.0.8	Ejercicio 8	7
1.0.9	Ejercicio 9	7
1.0.10	Ejercicio 10	7

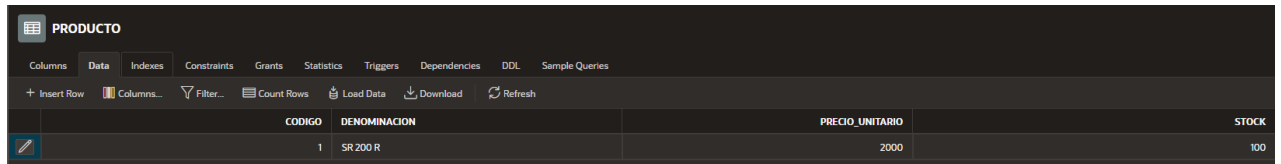
1 Lenguaje DML

1.0.1 Ejercicio 1

Consigna: Escribir 2 sentencias de INSERT (una precisando los campos y otra solo los valores) :

Listing 1: Sentencia Insert con campos.

```
1 insert into PRODUCTO(Codigo, Denominacion, Precio_Unitario, Stock)
2 values('00000001', 'SR 200 R', '2000', '100');
```



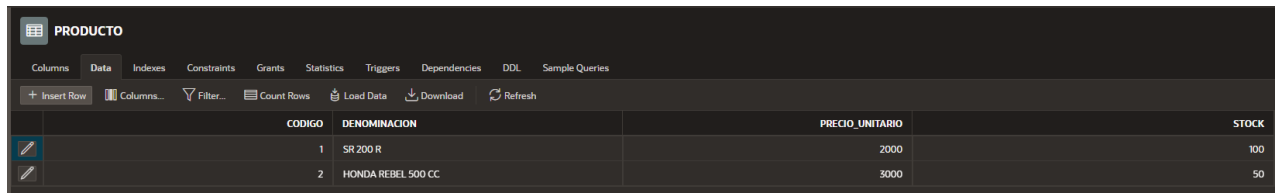
The screenshot shows a database interface for a table named 'PRODUCTO'. The interface includes tabs for Columns, Data, Indexes, Constraints, Grants, Statistics, Triggers, Dependencies, DDL, and Sample Queries. Below these tabs are buttons for '+ Insert Row', 'Columns...', 'Filter...', 'Count Rows', 'Load Data', 'Download', and 'Refresh'. The table has four columns: CODIGO, DENOMINACION, PRECIO_UNITARIO, and STOCK. A single row is displayed with the values: 1, SR 200 R, 2000, and 100.

	CODIGO	DENOMINACION	PRECIO_UNITARIO	STOCK
	1	SR 200 R	2000	100

Figure 1: Comprobación Insert con campos.

Listing 2: Sentencia Insert sin campos.

```
1 insert into PRODUCTO
2 values('00000002', 'HONDA REBEL 500 CC', '3000', '50');
```



The screenshot shows the same database interface for the 'PRODUCTO' table. It now contains two rows. The first row is the same as in Figure 1. The second row has the values: 2, HONDA REBEL 500 CC, 3000, and 50.

	CODIGO	DENOMINACION	PRECIO_UNITARIO	STOCK
	1	SR 200 R	2000	100
	2	HONDA REBEL 500 CC	3000	50

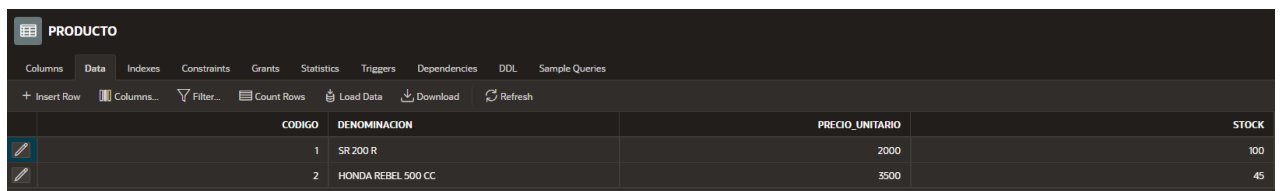
Figure 2: Comprobación Insert sin campos.

1.0.2 Ejercicio 2

Consigna: Escribir 1 sentencia de UPDATE.

Listing 3: Sentencia Update.

```
1 update PRODUCTO
2 set Precio_Unitario = '3500'
3 , Stock = '45'
4 where Codigo = '00000002';
```



The screenshot shows the database interface for the 'PRODUCTO' table. The second row has been updated. Its values are now: 2, HONDA REBEL 500 CC, 3500, and 45.

	CODIGO	DENOMINACION	PRECIO_UNITARIO	STOCK
	1	SR 200 R	2000	100
	2	HONDA REBEL 500 CC	3500	45

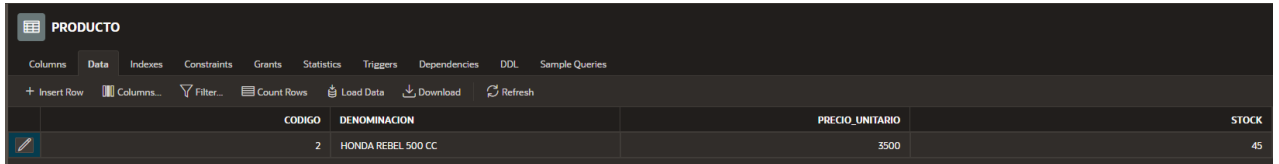
Figure 3: Comprobación sentencia Update.

1.0.3 Ejercicio 3

Consigna: Escribir 1 sentencia DELETE.

Listing 4: Sentencia Delete.

```
1 delete from PRODUCTO
2   where Codigo = '00000001';
```



CODIGO	DENOMINACION	PRECIO_UNITARIO	STOCK
2	HONDA REBEL 500 CC	3500	45

Figure 4: Comprobación sentencia Delete.

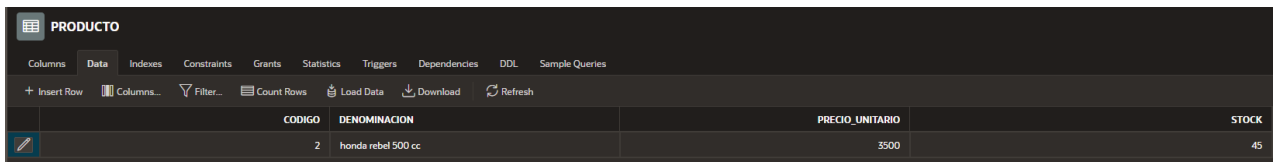
1.0.4 Ejercicio 4

Consigna: Escribir la sentencia que transforme el campo denominación a mayúscula de la tabla PRODUCTO.

Como ya lo tenía en mayúsculas (en el DB), lo actualizo a minúsculas y después paso a mayúsculas con un comando.

Listing 5: Paso a minus.

```
1 update PRODUCTO
2 set Denominacion = lower(Denominacion);
```



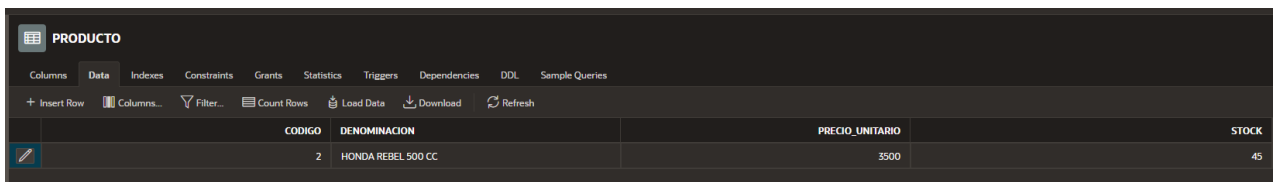
CODIGO	DENOMINACION	PRECIO_UNITARIO	STOCK
2	honda rebel 500 cc	3500	45

Figure 5: Comprobación minus.

Paso a mayúsculas el campo:

Listing 6: Paso a mayus.

```
1 update PRODUCTO
2 set Denominacion = upper(Denominacion);
```



CODIGO	DENOMINACION	PRECIO_UNITARIO	STOCK
2	HONDA REBEL 500 CC	3500	45

Figure 6: Comprobación mayus.

1.0.5 Ejercicio 5

Consigna: Escribir la sentencia que tome los primeros 5 caracteres del campo denominación de la tabla PRODUCTO.

Listing 7: Selecciono el substring dentro de denominación que va desde el primer carácter hasta el 5 inclusive.

```
1 select substr(Denominacion, 1, 5)
2 from PRODUCTO;
```

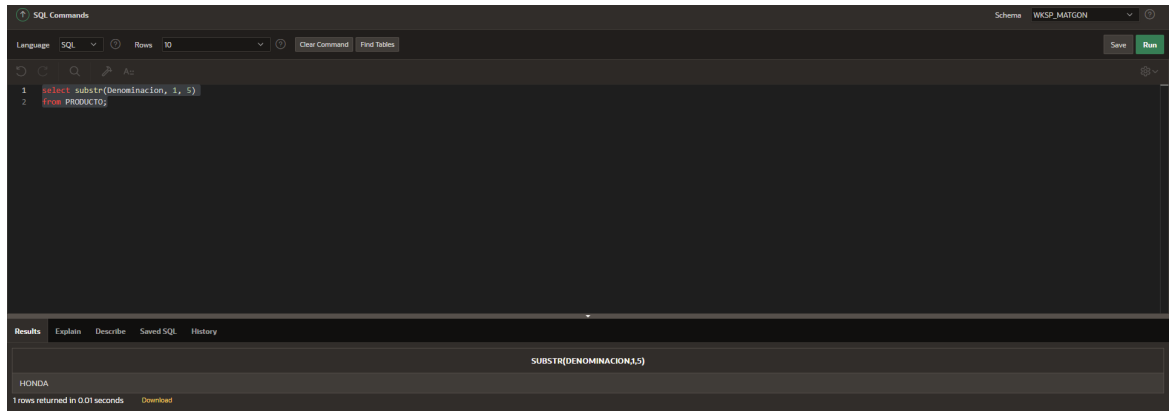


Figure 7: Comprobación Substr.

1.0.6 Ejercicio 6

Consigna: Escribir la sentencia que busque sobre la tabla PRODUCTO si existen registros cuya DENOMINACIÓN contenga la palabra “YOGUR”.

Le meto upper porque no sé en qué formato podría estar escrito (en realidad sé, pero para ser más exactos y/o prácticos).

Listing 8: Busco denominación igual a YOGUR en mi DB.

```
1 select DENOMINACION
2 from PRODUCTO
3 where upper(DENOMINACION) = 'YOGUR';
```



Figure 8: Resultado obtenido.

Contrasto con información conocida de mi DB:

Listing 9: Chequeo con info conocida.

```
1 select DENOMINACION
2 from PRODUCTO
3 where upper(DENOMINACION) = 'HONDA REBEL 500 CC';
```

DENOMINACION
HONDA REBEL 500 CC

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

Figure 9: Resultado obtenido.

Compruebo que el comando upper haya sido bien escrito:

Listing 10: Comando upper.

```

1 select DENOMINACION
2   from PRODUCTO
3  where DENOMINACION = 'honda rebel 500 cc';

```

no data found

Figure 10: Resultado obtenido de upper.

Veo que fue bien ejecutado.

1.0.7 Ejercicio 7

Consigna: Escribir la sentencia que cuente la cantidad de caracteres que tiene el campo DENOMINACIÓN, en todos los registros de la tabla PRODUCTO.

Listing 11: Pido la longitud del campo de la columna Denominación de producto.

```

1 select length(DENOMINACION)
2   from PRODUCTO;

```

length(DENOMINACION)
18

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

Figure 11: Comprobación de lenght.

Meto otra fila para que sea más notorio:

Listing 12: Inserto otra fila.

```

1 insert into PRODUCTO(Codigo, Denominacion, Precio_Unitario, Stock)
2   values('00000003', 'HONDA CBR 1000 RR', '3000', '90');

```

	CODIGO	DENOMINACION	PRECIO_UNITARIO	STOCK
	3	HONDA CBR 1000 RR	3000	90
	2	HONDA REBEL 500 CC	3500	45

Figure 12: Comprobación de nueva fila.

Aplico misma sentencia que antes:

Listing 13: Pido la longitud del campo de la columna Denominación de producto.

```
1 select length(DENOMINACION)
2 from PRODUCTO;
```



length(DENOMINACION)
17
18

2 rows returned in 0.01 seconds [Download](#)

Figure 13: Comprobación de nueva longitud.

1.0.8 Ejercicio 8

Consigna: Escribir la sentencia que busque si hay registros que tengan el campo DENOMINACIÓN sin valor.

Listing 14: Busco null en columna Denominación.

```
1 select DENOMINACION
2 from PRODUCTO
3 where upper(DENOMINACION) is null;
```



no data found

Figure 14: Resultado de anterior búsqueda.

1.0.9 Ejercicio 9

Consigna: Escribir la sentencia que busque los productos cuyo stock sea entre 0 y 1000.

Listing 15: Busco los productos stock pero con el filtro de entre 0 y 1000 unidades.

```
1 select *
2 from PRODUCTO
3 where 0 < STOCK and STOCK < 1000;
```



	CODIGO	DENOMINACION	PRECIO_UNITARIO	STOCK
5		HONDA CBR 1000 RR	5000	90
2		HONDA REBEL 500 CC	3500	45

2 rows returned in 0.01 seconds [Download](#)

Figure 15: Resultado del filtro en stock de 0 a 1000 unidades.

1.0.10 Ejercicio 10

Consigna: Escribir la sentencia que traiga los productos cuya denominación este en estos valores ('YOGUR','LECHE').

Listing 16: Busco los productos cuya denominación sea igual a algunos de los dos strings presentados en la consigna.

```
1 select *
2   from PRODUCTO
3  where DENOMINACION = 'YOGUR' or DENOMINACION = 'LECHE';
```



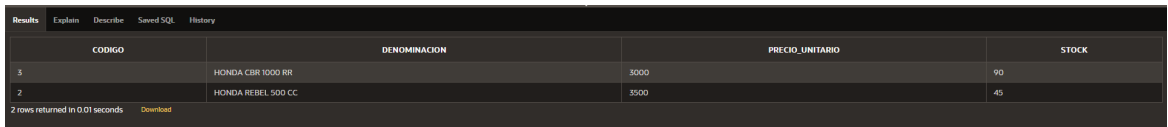
The screenshot shows a database interface with a dark theme. At the top, there are tabs: 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is selected. Below the tabs, the text 'no data found' is displayed in a light gray font.

Figure 16: Resultado del filtro en denominación anterior.

Compruebo sintaxis con algo que sé que funcionará.

Listing 17: Busco los productos cuya denominación sea igual a algunos de los dos strings que sé que están en mi DB.

```
1 select *
2   from PRODUCTO
3  where DENOMINACION = 'HONDA CBR 1000 RR' or DENOMINACION = 'HONDA
   REBEL 500 CC';
```



The screenshot shows a database interface with a dark theme. At the top, there are tabs: 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is selected. Below the tabs, a table with 4 columns is displayed. The columns are labeled 'CODIGO', 'DENOMINACION', 'PRECIO_UNITARIO', and 'STOCK'. There are two rows of data. The first row has CODIGO 3, DENOMINACION 'HONDA CBR 1000 RR', PRECIO_UNITARIO 3000, and STOCK 90. The second row has CODIGO 2, DENOMINACION 'HONDA REBEL 500 CC', PRECIO_UNITARIO 3500, and STOCK 45. At the bottom left, it says '2 rows returned in 0.01 seconds' and there is a 'Download' link.

CODIGO	DENOMINACION	PRECIO_UNITARIO	STOCK
3	HONDA CBR 1000 RR	3000	90
2	HONDA REBEL 500 CC	3500	45

Figure 17: Resultado del filtro en denominación anterior.