

## RESPUESTAS A LA TAREA DE LA SECCION UNIENDO DIFERENTES TABLAS

1.- Revisa el video de la lectura llamada “Inner Join” para que prepares la base de datos que vamos a analizar, y, mediante la unión de las tablas “Datos\_Personales” y “Actividades\_Extra”, obtén una lista de los nombres, los correos y la Clase Extra que están tomando los alumnos que se encuentran registrados en alguna de las Actividades Extra-Curriculares. Ordena la lista de forma ascendente en base a la Actividad en la que se encuentran registrados los alumnos

### RESPUESTA:

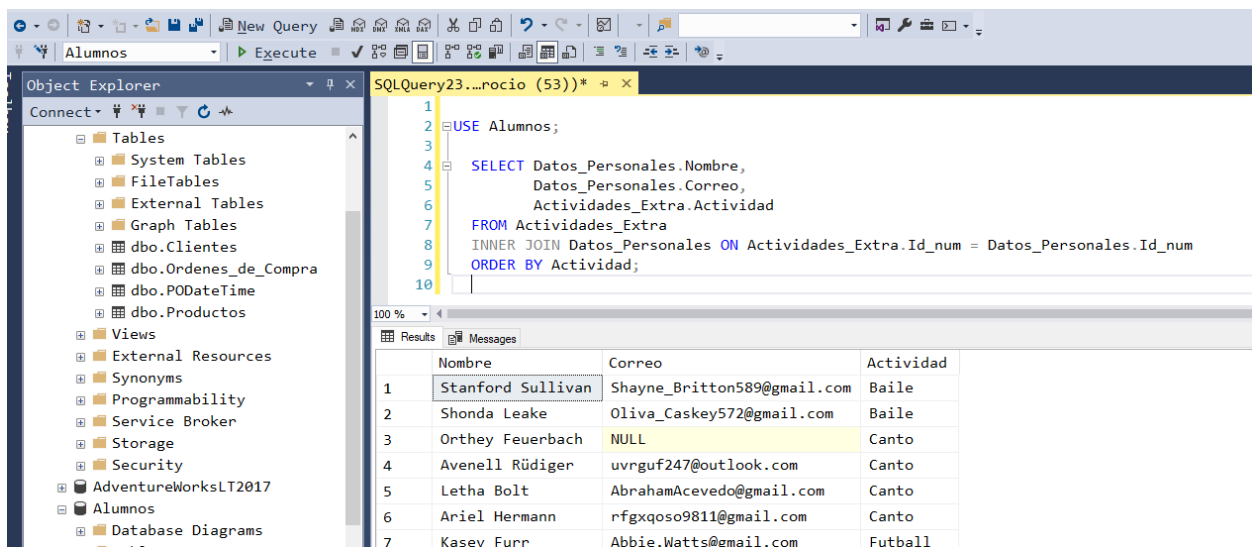
```
USE Alumnos;
```

```
SELECT Datos_Personales.Nombre,  
       Datos_Personales.Correo,  
       Actividades_Extra.Actividad
```

```
FROM Actividades_Extra
```

```
INNER JOIN Datos_Personales ON Actividades_Extra.Id_num = Datos_Personales.Id_num
```

```
ORDER BY Actividad;
```



The screenshot shows the SQL Server Enterprise Manager interface. On the left, the Object Explorer displays the database structure, including tables like 'dbo.Clientes', 'dbo.Ordenes\_de\_Compra', 'dbo.PODateTime', and 'dbo.Productos'. The main window displays a SQL query in the 'SQLQuery23...rocio (53)\*' window. The query is as follows:

```
1  
2 USE Alumnos;  
3  
4 SELECT Datos_Personales.Nombre,  
5        Datos_Personales.Correo,  
6        Actividades_Extra.Actividad  
7 FROM Actividades_Extra  
8 INNER JOIN Datos_Personales ON Actividades_Extra.Id_num = Datos_Personales.Id_num  
9 ORDER BY Actividad;  
10
```

Below the query, the 'Results' tab shows the output of the query. The results are displayed in a table with three columns: 'Nombre', 'Correo', and 'Actividad'. The data is sorted by 'Actividad' in ascending order.

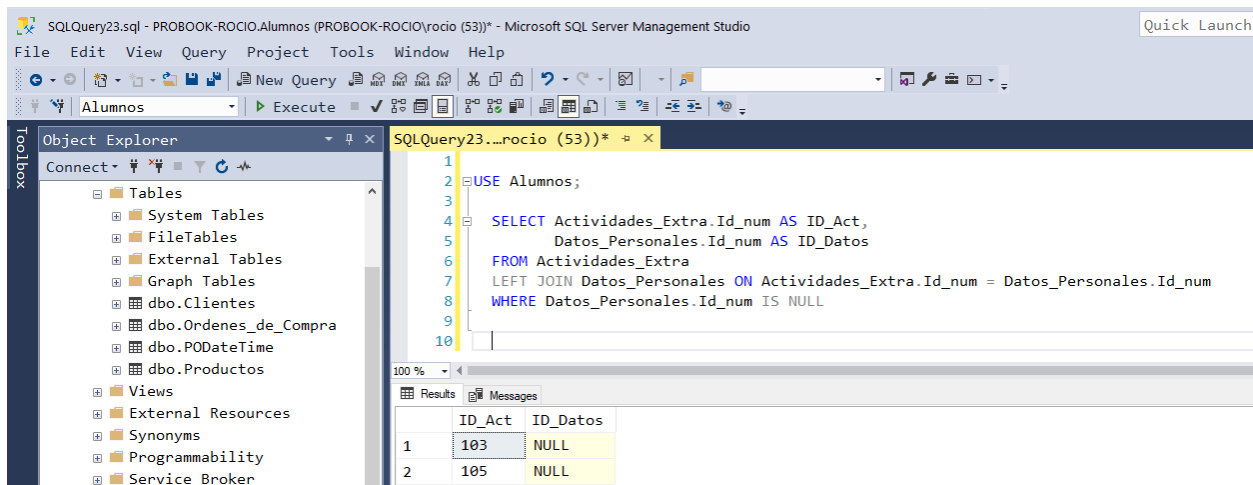
	Nombre	Correo	Actividad
1	Stanford Sullivan	Shayne_Britton589@gmail.com	Baile
2	Shonda Leake	Oliva_Caskey572@gmail.com	Baile
3	Orthey Feuerbach	NULL	Canto
4	Avenell Rüdiger	uvrguf247@outlook.com	Canto
5	Letha Bolt	AbrahamAcevedo@gmail.com	Canto
6	Ariel Hermann	rfgxqoso9811@gmail.com	Canto
7	Kasey Furr	Abbie.Watts@gmail.com	Futball

2.- Utilizando las Tablas Actividades\_Extra y Datos\_Personales, obtén una lista de los Números de Identificación de los alumnos que están incluidos en la lista de Actividades Extra-curriculares pero que no se encuentran incluidos en la lista de alumnos de la Tabla Datos\_personales

**RESPUESTA:**

```
USE Alumnos;
```

```
SELECT Actividades_Extra.Id_num AS ID_Act,  
       Datos_Personales.Id_num AS ID_Datos  
FROM Actividades_Extra  
LEFT JOIN Datos_Personales ON Actividades_Extra.Id_num = Datos_Personales.Id_num  
  
WHERE Datos_Personales.Id_num IS NULL
```



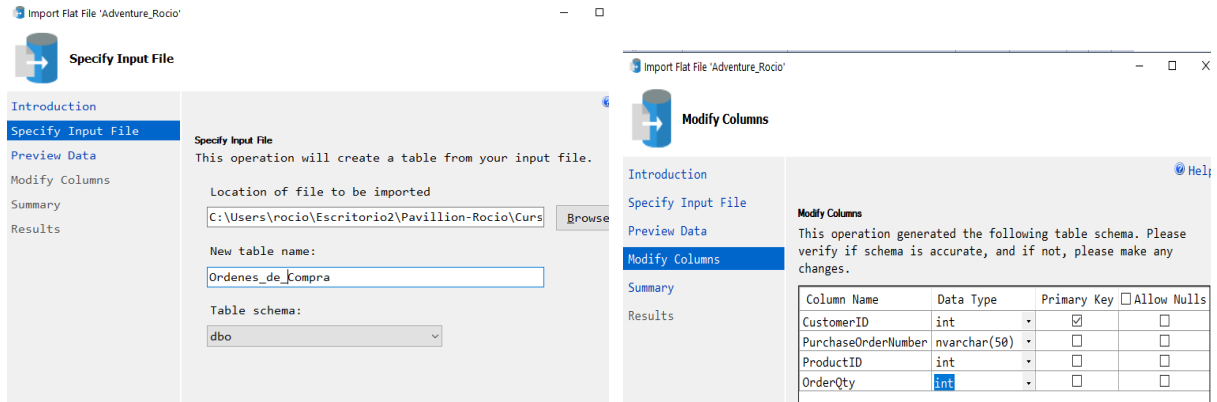
The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar indicates the file is 'SQLQuery23.sql - PROBOOK-ROCIO.Alumnos (PROBOOK-ROCIO\rocio (53))' and the server is 'Microsoft SQL Server Management Studio'. The menu bar includes File, Edit, View, Query, Project, Tools, Window, and Help. The toolbar contains various icons for file operations, query execution, and formatting. The Object Explorer on the left shows the database structure, including Tables (System Tables, FileTables, External Tables, Graph Tables, dbo.Cientes, dbo.Ordenes\_de\_Compra, dbo.PODateTime, dbo.Productos), Views, External Resources, Synonyms, Programmability, and Service Broker. The query editor in the center contains the following SQL code:

```
1  
2 USE Alumnos;  
3  
4 SELECT Actividades_Extra.Id_num AS ID_Act,  
5        Datos_Personales.Id_num AS ID_Datos  
6 FROM Actividades_Extra  
7 LEFT JOIN Datos_Personales ON Actividades_Extra.Id_num = Datos_Personales.Id_num  
8 WHERE Datos_Personales.Id_num IS NULL  
9  
10
```

The Results pane at the bottom shows the output of the query, which consists of two rows of data:

	ID_Act	ID_Datos
1	103	NULL
2	105	NULL

3.- Si todavía no has cargado la información de las tablas Clientes, Ordenes\_de\_Compra y Productos, de la base de datos que creamos en el video “Cargando un archivo csv en SQL Server”, es momento de hacerlo. Ten especial cuidado al momento de definir el formato de las columnas y asegúrate de que OrderQty se cargue como una variable numérica.



4.- Utilizando las tablas Clientes, Ordenes\_de\_Compra y Productos, de la base de datos que creamos en el video “Cargando un archivo csv en SQL Server”, obtén una lista de los productos adquiridos por cada uno de los Clientes, incluyendo sus descripciones y las cantidades ordenadas

**RESPUESTA:**

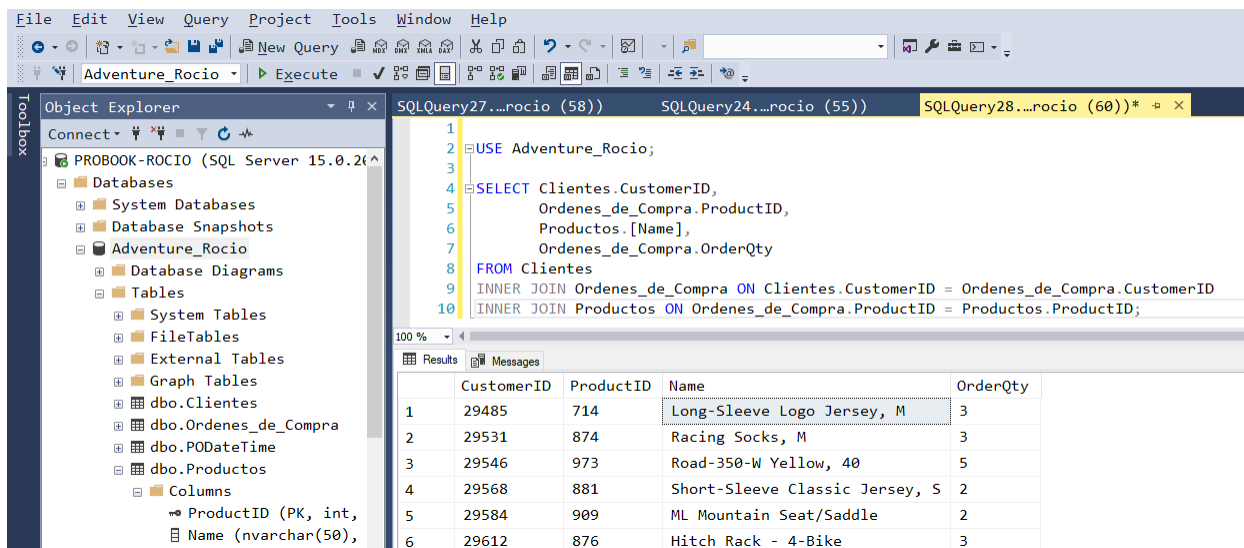
```
USE Adventure_Rocio;
```

```
SELECT Clientes.CustomerID,  
       Ordenes_de_Compra.ProductID,  
       Productos.[Name],  
       Ordenes_de_Compra.OrderQty
```

```
FROM Clientes
```

```
INNER JOIN Ordenes_de_Compra ON Clientes.CustomerID = Ordenes_de_Compra.CustomerID
```

```
INNER JOIN Productos ON Ordenes_de_Compra.ProductID = Productos.ProductID;
```



The screenshot shows the SQL Server Enterprise Manager interface. On the left, the Object Explorer displays the database structure for 'PROBOOK-ROCIO (SQL Server 15.0.2605.1)'. The 'Adventure\_Rocio' database is expanded, showing tables: 'dbo.Clientes', 'dbo.Ordenes\_de\_Compra', 'dbo.PODateTime', and 'dbo.Productos'. The 'dbo.Productos' table is selected, showing its columns: 'ProductID (PK, int)' and 'Name (nvarchar(50))'. The main window displays a SQL query in the 'SQLQuery27.rocio (58)' tab. The query is as follows:

```
1  
2 USE Adventure_Rocio;  
3  
4 SELECT Clientes.CustomerID,  
5        Ordenes_de_Compra.ProductID,  
6        Productos.[Name],  
7        Ordenes_de_Compra.OrderQty  
8 FROM Clientes  
9 INNER JOIN Ordenes_de_Compra ON Clientes.CustomerID = Ordenes_de_Compra.CustomerID  
10 INNER JOIN Productos ON Ordenes_de_Compra.ProductID = Productos.ProductID;
```

Below the query, the 'Results' tab shows the output of the query. The results are displayed in a table with the following columns: 'CustomerID', 'ProductID', 'Name', and 'OrderQty'. The results are as follows:

	CustomerID	ProductID	Name	OrderQty
1	29485	714	Long-Sleeve Logo Jersey, M	3
2	29531	874	Racing Socks, M	3
3	29546	973	Road-350-W Yellow, 40	5
4	29568	881	Short-Sleeve Classic Jersey, S	2
5	29584	909	ML Mountain Seat/Saddle	2
6	29612	876	Hitch Rack - 4-Bike	3

5.- Utiliza la información obtenida en el punto anterior para saber cuantas unidades se vendieron de cada uno de los productos

**RESPUESTA:**

Crea una vista con la consulta del punto anterior y corre el código para que se forme la tabla temporal “Productos\_Vendidos”

```
CREATE VIEW Productos_Vendidos AS

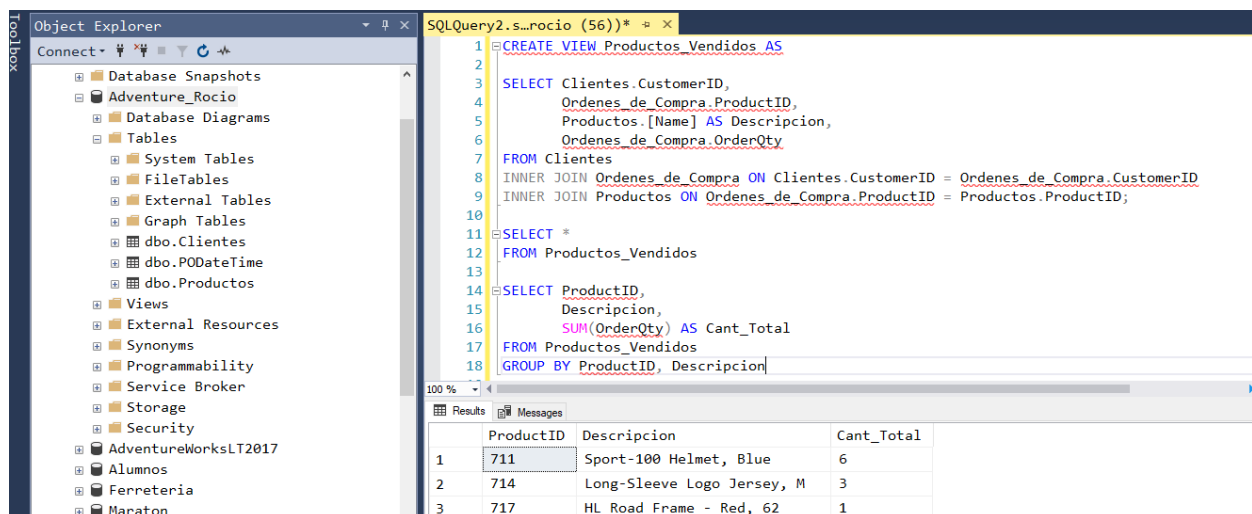
SELECT Clientes.CustomerID,
       Ordenes_de_Compra.ProductID,
       Productos.[Name] AS Descripcion,
       Ordenes_de_Compra.OrderQty
FROM Clientes
INNER JOIN Ordenes_de_Compra ON Clientes.CustomerID = Ordenes_de_Compra.CustomerID
INNER JOIN Productos ON Ordenes_de_Compra.ProductID = Productos.ProductID;
```

Si así lo deseas, puedes ver los campos de la tabla temporal que acabas de crear

```
SELECT *
FROM Productos_Vendidos
```

Una vez creada la tabla, realiza otra consulta sumando la cantidad de productos vendidos

```
SELECT ProductID,
       Descripcion,
       SUM(OrderQty) AS Cant_Total
FROM Productos_Vendidos
GROUP BY ProductID, Descripcion
```



	ProductID	Descripcion	Cant_Total
1	711	Sport-100 Helmet, Blue	6
2	714	Long-Sleeve Logo Jersey, M	3
3	717	HL Road Frame - Red, 62	1