

```

-- G450_PIRIZ_P4_JOINS
/*
1. Obtener el ID de persona y el nombre de todas las personas que son jefes.
*/
SELECT
    p.BusinessEntityID,
    p.FirstName,
    p.LastName
FROM
    HumanResources.Employee AS e
    INNER JOIN Person.Person AS p on e.BusinessEntityID = p.BusinessEntityID
WHERE
    e.JobTitle LIKE 'chief%'
;
--
-----
--
--
-----
--
/*
2. Obtener el ID de persona, el nombre de la persona y el mail de cada uno de ellos.
*/
SELECT
    p.BusinessEntityID,
    p.FirstName,
    p.LastName,
    e.EmailAddress
FROM
    Person.Person AS p
    INNER JOIN Person.EmailAddress AS e on p.BusinessEntityID = e.BusinessEntityID
;
--
-----
--
--
-----
--
/*
3. Obtener el ID de persona,
el nombre de persona y
el teléfono de cada persona
que su apellido comience con 'J'.
*/
SELECT
    p.BusinessEntityID,
    p.FirstName,
    p.LastName,
    pp.PhoneNumber
FROM
    Person.Person AS p
    INNER JOIN Person.PersonPhone AS pp on p.BusinessEntityID = pp.BusinessEntityID
WHERE
    P.LastName LIKE 'J%'
;
--
-----
--
--
-----
--
/*
4. Obtener el ID de producto,
el nombre del producto
y la descripción de la subcategoría de cada producto de color ROJO, AZUL o Negro

```

solamente de los productos que TIENEN SUBCATEGORIA.

*/

SELECT

p.ProductID,
p.Name,
ps.Name

FROM

Production.Product AS p

INNER JOIN Production.ProductSubcategory AS ps on p.ProductSubcategoryID =

ps.ProductSubcategoryID

WHERE

p.Color in ('RED', 'BLUE', 'BLACK')

;

--

--

--

--

/*

5. Obtener el ID de producto,

el nombre del producto

y la descripción de la subcategoría de TODOS LOS PRODUCTOS.

*/

SELECT

p.ProductID,
p.Name,
ps.Name

FROM

Production.Product AS p

LEFT JOIN Production.ProductSubcategory AS ps on p.ProductSubcategoryID =

ps.ProductSubcategoryID

;

--

--

--

--

/*

6. Obtener un listado que muestre

el ID de cliente,

el ID del store

y el nombre de este (concatenado en un mismo campo) de todos los clientes.

*/

SELECT

CONCAT(
C.CustomerID, ', ',
C.StoreID, ', ',
S.Name)

FROM

Sales.Customer as C

join Sales.Store AS S ON C.StoreID = S.BusinessEntityID

;

--

--

--

--

/*

7. Obtener un listado de todos los clientes que contenga:

el ID de cliente y el nombre de cada uno de los territorios

donde compro cada cliente.

*/

```

SELECT
    SOH.CustomerID,
    ST.Name
FROM
    Sales.SalesOrderHeader AS SOH
    LEFT JOIN Sales.SalesTerritory AS ST ON SOH.TerritoryID = ST.TerritoryID;
--
--
--
--
/*
8. Agrupar el reporte anterior
y mostrar la cantidad de clientes que existe en
cada territorio y ordenarlos de mayor a menor.
*/
SELECT
    COUNT(SOH.CustomerID) AS CANT_CLIENTES,
    ST.Name
FROM
    Sales.SalesOrderHeader AS SOH
    LEFT JOIN Sales.SalesTerritory AS ST ON SOH.TerritoryID = ST.TerritoryID
GROUP BY
    ST.Name
ORDER BY
    1 DESC;
--
--
--
--
/*
9. Obtener un listado que muestre el ID de cliente,
el nombre del store
y el nombre del territorio de los clientes.
*/
SELECT
    C.CustomerID,
    S.Name,
    ST.Name
FROM
    Sales.Customer AS C
    INNER JOIN Sales.Store as S ON C.StoreID = S.BusinessEntityID
    INNER JOIN Sales.SalesTerritory AS ST ON C.TerritoryID = ST.TerritoryID
;
--
--
--
--
/*
10. La gerencia de AdventureWorks está tratando de aprender más acerca de sus clientes
y por ello le gustaría contactar a cada uno
mediante un email o un llamado telefónico para realizar una pequeña encuesta.
Te ha tocado a ti realizar un listado que permita este tipo de acción.
Las principales tareas en este ejercicio son:
• Identificar que tabla contiene esta información
• Elegir los campos que contienen la información deseada
• Generar una consulta usando el comando SELECT, FROM y el tipo de JOIN adecuado
*/
SELECT
    C.CustomerID,

```

```

P.FirstName,
P.LastName,
PE.EmailAddress,
PP.PhoneNumber
FROM
Sales.Customer as C
INNER JOIN Person.Person P ON C.PersonID = P.BusinessEntityID
INNER JOIN Person.EmailAddress as PE ON P.BusinessEntityID = PE.BusinessEntityID
INNER JOIN Person.PersonPhone AS PP ON P.BusinessEntityID = PP.BusinessEntityID;

```

```

--
-----
--
--
-----
--

```

```

/*
11. Obtener el nombre, el apellido y el número de tarjeta de crédito
de todas las personas que existen en la base de datos.
Aclaración: se quiere ver en el reporte a todas las personas (tengan o no tarjeta).
• ¿Qué tablas participan del reporte?
• Realice el reporte mediante una consulta SQL
*/

```

```

SELECT
P.FirstName,
P.LastName,
CC.CardNumber
FROM
Person.Person AS P
LEFT JOIN Sales.PersonCreditCard AS PCC ON P.BusinessEntityID = PCC.BusinessEntityID
LEFT JOIN Sales.CreditCard AS CC ON PCC.CreditCardID = CC.CreditCardID

```