* Nos Conectamos como administradores con:

***conn sys as sysdba***

* Creamos un usario llamado c##jardineria y le concedemos los permisos.

***create user c##jardineria identified by jardineria default tablespace users;***

***grant connect, resource,DBA to c##jardineria;***

***CONNECT c##jardineria/jardineria;***

Para la base de datos jardinería (jardineria.sql).

DQL

1. Defina las siguientes consultas:
2. Sacar el código de oficina y la ciudad donde hay oficinas.
3. Cuantos empleados hay en la compañía.
4. Cuantos clientes tiene cada país.
5. Pago medio en 2005.
6. Cuantos pedidos están en cada estado ordenado descendente por el número de pedidos.
7. Obtener el precio del producto más caro y del más barato.
8. Obtener el nombre del cliente con mayor límite de crédito.
9. Obtener el nombre, apellido1 y cargo de los empleados que no representen a ningún cliente.
10. Obtener el nombre del producto más caro.
11. Obtener el nombre del producto del que más unidades se han vendido en un mismo pedido.
12. Obtener los clientes cuya línea de crédito sea mayor que los pagos que haya realizado.
13. Sacar el producto que más unidades tiene en stock y el que menos unidades tiene en stock.
14. Numero de productos que se piden en cada pedido.
15. Listado con el nombre de cada cliente y el nombre y apellido de su representante de ventas.
16. Nombre de los clientes que no han realizado pagos junto con el nombre de representante.
17. Nombre, unidades vendidas, total facturado y total con impuestos de los productos que hayan facturado más de 3000 euros.
18. Dirección de las oficinas que tengan clientes de Fuenlabrada.
19. Listado con el precio total de cada pedido.
20. Cliente que hizo el pago con mayor cuantía y el que hizo el pago con menor cuantía.
21. Clientes que hayan hecho pedidos en el 2008 por una cuantía superior a 2000 euros.
22. ¿Pedido más caro del empleado que más clientes tiene?.
23. Ciudad y teléfono de las oficinas de EEUU.
24. Nombre, apellidos y email de los empleados a cargo de Alberto Soria.
25. Nombre, apellidos y cargo de aquellos que no sean representantes de ventas.
26. Nombre de los clientes españoles
27. Número de clientes de las ciudades que empiezan por M.
28. Código de empleado y el número de clientes al que atiende cada representante de ventas.
29. Número de clientes que no tiene asignado representante de ventas.
30. Código de cliente de aquellos clientes que hicieron pagos en 2008.
31. Distintos estados por los que puede pasar un pedido.
32. Numero de pedido, código de cliente, fecha requerida y fecha de entrega de los pedidos cuya fecha de entrega ha sido al menos dos días antes de la fecha requerida.
33. Facturación que ha tenido la empresa en toda la historia, indicando la base imponible, el IVA y el total facturado. Nota: La base imponible se calcula sumando el coste del producto por el número de unidades vendidas. El IVA es el 21% de la base imponible.
34. Listado con el nombre de los empleados junto con el nombre de sus jefes.
35. Listado de clientes indicando el nombre del cliente y cuantos pedidos ha realizado.
36. Listado con los nombres de los clientes y el total pagado por cada uno de ellos.
37. Cuantos empleados tiene cada oficina, mostrando el nombre de la ciudad donde está la oficina.
38. Nombre, apellido, oficina (ciudad) y cargo del empleado que no represente a ningún cliente.
39. Media de unidades de stock de los productos agrupados por gama.
40. Clientes que residan en la misma ciudad donde hay una oficina, indicando dónde está la oficina.
41. Clientes que residan en ciudades donde no hay oficinas ordenado por la ciudad donde residen.

DML

1. Crea y ejecuta un script ‘actualiza.sql’ que realice las siguientes acciones:
2. Inserta una oficina con sede en Fuenlabrada
3. Inserta un empleado para la oficina de Fuenlabrada que sea representante de ventas
4. Inserta un cliente del representante de ventas en el punto 2.
5. Inserta un pedido del cliente anterior (con su detalle) de al menos 2 productos.
6. Actualiza el código del cliente insertado y averigua si hubo cambios en las tablas relacionadas.
7. Borra el cliente y verifica si hubo cambios
8. Usando subconsultas en los filtros, realiza las siguientes actualizaciones:

1.Borra los clientes que no tengan pedidos

2.Incrementa en un 20% el precio de los productos

3.Borra los pagos del cliente con menor límite de credito.

4.Establece a 0 el límite de crédito del cliente que menos unidades pedidas tenga del producto ‘OR-179’

PL/SQL

1. Triggers

Cree un disparador para que cuando se borre un cliente de la base de datos, se almacene en una tabla que ya existirá previamente, que se llame “clientes\_borrados” con la siguiente información: Nombre\_cliente,Fecha\_borrado,Importe\_facturación.

1. Procedimientos

Deseamos tener un procedimiento que pasemos **el código de un cliente** y nos liste los datos de ese cliente: Código, Nombre, Ciudad y País, así como los pagos que ha realizado, ordenados cronológicamente. Para finalizar que muestre la cantidad total pagada. Fíjate bien en la imagen capturada para ver todos los detalles que debe mostrar el procedimiento. Implementa también el tratamiento de excepciones.

Tabla

Descripción generada automáticamente