**PRUEBA TECNICA DE CONOCIMIENTOS SQL (PL/SQL – Transact-SQL) .NET**

**Nombre**: \_\_Henderson Yhardany Sotelo Sepúlveda\_\_\_\_

**Fecha**: \_\_26/04/2021\_\_

**Car Center**

**Servicio de Mantenimiento:**

CAR CENTER es un taller que atiende a sus clientes a nivel nacional, prestando el servicio de reparación de vehículos con mecánicos altamente calificados y proporcionando todos los repuestos necesarios en el mantenimiento.

El taller provee a sus CLIENTES una plataforma donde pueden registrarse e inscribir sus VEHICULOS. De esta manera, cuando el cliente requiere un mantenimiento, puede solicitarlo a través de la plataforma e incluso subir fotos del estado de su vehículo para que pueda iniciarse una valoración del arreglo.

**Facturación:**

Dado el número de identificación del CLIENTE se genera una FACTURA para el cliente, la cual incluye el cobro de todos los MANTENIMIENTOS es estado “*terminado*”. La FACTURA debe contener todos los datos del cliente (primer nombre, segundo nombre, primer apellido, segundo apellido, , tipo de documento, documento, celular, dirección y correo electrónico), los datos del mecánico que hizo el mantenimiento (primer nombre, segundo nombre, primer apellido, segundo apellido, , tipo de documento, documento, celular, dirección y correo electrónico, estado), el desglose de los repuestos utilizados (precio por unidad, número de unidades, descuento), el desglose de servicios (precio de la mano de obra, descuento), y el total de la factura.

A este total se le aplica el 19 % de IVA.

En la facturación se debe tener en cuenta:

- Se aplica un descuento del 50% sobre los servicios de mano de obra cuando el valor total de los repuestos sea superior a $3.000.000.

- El valor de mano de obra de cada uno de los tipos de servicio tiene un valor mínimo y máximo definido.

- No se debe emitir facturas que superen el presupuesto que estableció el CLIENTE (en caso de que él haya establecido ese límite).

En este contexto se necesita crear un CRUD (crear, consultar, actualizar, y eliminar) para la funcionalidad de clientes:

**Entregables:**

1. Definición de Módulos
2. Tablas a crear con columnas y tipos
3. Definición de llaves Primarias y Foráneas
4. Modelo E/R
5. Consulta de Clientes que han comprado un acumulado $100.000 en los últimos 60 días

SELECT C.ID CLIENTE\_ID, C.PRIMER\_NOMBRE || ' ' || C.SEGUNDO\_NOMBRE || ' ' || C.PRIMER\_APELLIDO || ' ' || C.SEGUNDO\_APELLIDO NOMBRE\_CLIENTE

FROM(

SELECT A.ID\_FACTURA

FROM

(SELECT FA.ID ID\_FACTURA, SUM(FA.VALOR) TOTAL

FROM HYSOTELO.FACTURA FA

WHERE FA.ESTADO = 'A'

AND TO\_DATE(FA.FECHA\_EMISION,'DD/MM/YYYY') >= (SELECT TO\_DATE(SYSDATE-60,'DD/MM/YYYY') FROM DUAL)

GROUP BY FA.ID)A

WHERE A.TOTAL = 100000)B, HYSOTELO.MANTENIMIENTO M, HYSOTELO.CLIENTE C

WHERE M.ID = B.ID\_FACTURA AND M.CLIENTE\_ID = C.ID

1. Consulta de los 100 productos más vendidos en los últimos 30 días

SELECT A.PRODUCTO\_ID, SUM(A.UNIDADES) CANTIDAD

FROM

(SELECT DISTINCT DM.PRODUCTO\_ID , DM.UNIDADES

FROM

(SELECT DISTINCT FA.ID, FA.MANTENIMIENTO\_ID

FROM HYSOTELO.FACTURA FA

WHERE FA.ESTADO = 'A' AND TO\_DATE(FA.FECHA\_EMISION,'DD/MM/YYYY') >= (SELECT TO\_DATE(SYSDATE-30,'DD/MM/YYYY') FROM DUAL))A, HYSOTELO.MANTENIMIENTO M,HYSOTELO.DETALLE\_MANTENIMIENTO DM

WHERE A.MANTENIMIENTO\_ID = M.ID AND DM.MANTENIMIENTO\_ID = M.ID

GROUP BY DM.PRODUCTO\_ID, DM.UNIDADES)A

WHERE ROWNUM <= 100

GROUP BY A.PRODUCTO\_ID

ORDER BY CANTIDAD DESC

1. Consulta de las tiendas que han vendido más de 100 UND del producto 100 en los últimos 60 días.

SELECT A.SEDE\_ID, (SELECT S.NOMBRE FROM HYSOTELO.SEDE S WHERE S.ID = A.SEDE\_ID)NOMBRE\_SEDE, SUM(A.UNIDADES)TOTAL\_UNIDADES\_VENDIDAS

FROM

(SELECT DISTINCT FA.ID, DM.UNIDADES, M.SEDE\_ID SEDE\_ID

FROM HYSOTELO.FACTURA FA, HYSOTELO.MANTENIMIENTO M, HYSOTELO.DETALLE\_MANTENIMIENTO DM

WHERE FA.ESTADO = 'A' AND

FA.MANTENIMIENTO\_ID = M.ID AND M.ID = DM.MANTENIMIENTO\_ID AND DM.ESTADO = 'A' AND DM.PRODUCTO\_ID = 100 AND

TO\_DATE(FA.FECHA\_EMISION,'DD/MM/YYYY') >= (SELECT TO\_DATE(SYSDATE-60,'DD/MM/YYYY') FROM DUAL))A

GROUP BY SEDE\_ID;

1. Consulta de todos los clientes que han tenido más de un(1) mantenimiento en los últimos 30 días.

SELECT A.CLIENTE\_ID, (SELECT C.PRIMER\_NOMBRE || ' ' || C.PRIMER\_APELLIDO FROM HYSOTELO.CLIENTE C WHERE C.ID = A.CLIENTE\_ID)NOMBRE\_CLIENTE ,A.TOTAL\_MANTENIMIENTOS

FROM

(SELECT DISTINCT M.CLIENTE\_ID, COUNT(M.ID) TOTAL\_MANTENIMIENTOS

FROM HYSOTELO.FACTURA FA, HYSOTELO.MANTENIMIENTO M

WHERE FA.ESTADO = 'A' AND TO\_DATE(FA.FECHA\_EMISION,'DD/MM/YYYY') >= (SELECT TO\_DATE(SYSDATE-30,'DD/MM/YYYY') FROM DUAL)

AND M.ID = FA.MANTENIMIENTO\_ID AND M.ESTADO = 'TERMINADO'

GROUP BY M.CLIENTE\_ID)A

1. Procedimiento que reste la cantidad de productos del inventario de las tiendas cada que se presente una venta.

CREATE OR REPLACE PROCEDURE HYSOTELO.DISMINUYE\_CANTIDAD\_PRODUCTOS(PRODUCTO\_ID NUMBER,CANTIDAD NUMBER, SALIDA OUT VARCHAR2)

IS

ID\_REG\_PROD NUMBER; *--ESTABLECE EL ID DEL REGISTRO DEL PRODUCTO A BUSCAR*

CANTIDAD\_DISP NUMBER;

BEGIN

SALIDA := 'ERROR';

*-- SE DEBE HACER LA CONSULTA DEL REGISTRO DEL PRODUCTO QUE LLEGA POR PARMETRO*

SELECT COUNT(P.ID) INTO ID\_REG\_PROD

FROM HYSOTELO.PRODUCTO P

WHERE P.ID = PRODUCTO\_ID AND P.ESTADO = 'A';

IF ID\_REG\_PROD >0 THEN *-- SI EXISTE EL REGISTRO SE CONSULTA LA CANTIDAD ACTUAL*

SELECT P.ID, P.CANTIDADES\_DISPONIBLES INTO ID\_REG\_PROD, CANTIDAD\_DISP

FROM HYSOTELO.PRODUCTO P

WHERE P.ID = PRODUCTO\_ID AND P.ESTADO = 'A';

UPDATE HYSOTELO.PRODUCTO P

SET P.CANTIDADES\_DISPONIBLES = CANTIDAD\_DISP - CANTIDAD

WHERE P.ID = ID\_REG\_PROD;

COMMIT;

SALIDA := 'OK';

ELSE

SALIDA := 'ERROR, NO EXISTE EL REGISTRO DE PRODUCTO';

END IF;

END DISMINUYE\_CANTIDAD\_PRODUCTOS;

**Convenciones**

* Lenguaje : Oracle PL/SQL ó Transact-SQL

**Por favor tener en cuenta:**

* Subir solución (Back-front-scripts) en repositorio Git y adjuntar URL

**Conceptos**

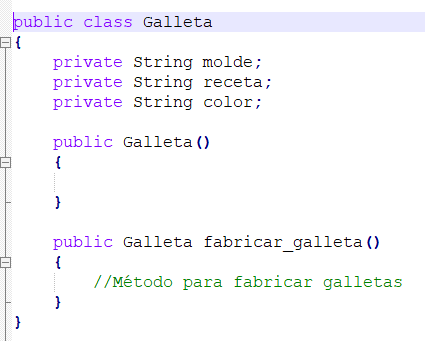
1. ¿Cuál es Componente básico para la ejecución de un programa en .Net?
2. Eclipse
3. Framework
4. Compilador
5. Ninguna de las anteriores
6. Describa de manera sintética:
7. Que es una clase

Rta: Una clase es una herramienta que permite crear un modelo computacional de un objeto, es como una especie de plantilla que sirve para crear objetos con una estructura predefinida. Las clases se usan para representar conceptos o entidades. Están formadas principalmente por lo atributos y los métodos que operan dichos atributos.

1. Que es un objeto:

Rta: Un objeto es una representación dentro de un modelo computacional que tiene un estado y un comportamiento definido, se pueden crear por medio de instancias a clases, descripción directa o replica de otros objetos. Los objetos se pueden comunicar entre sí ya que poseen una serie de dato almacenados y una serie de tareas definidas a ejecutar.

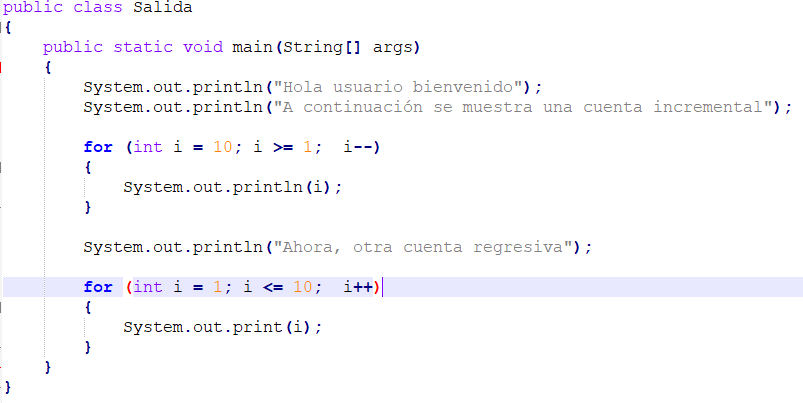
1. En el siguiente ejemplo de código:



el método Galleta es:

* 1. El método principal de la clase
  2. Un singleton de la clase
  3. El constructor de la clase
  4. Ninguno de los anteriores

1. en el siguiente ejemplo de código:



la salida del primer loop es:

1. "a,b,c,d,e,f,g,h,i,j"
2. "2,4,6,8,10,12,14,16,18,20"
3. "1,2,3,4,5,6,7,8,9,10"
4. "10,9,8,7,6,5,4,3,2,1"
5. ninguna de las anteriores
6. ¿Qué es un API?

Una API es un conjunto de definiciones y protocolos que buscan agilidad en el diseño, ya que no se requiere saber cómo están implementados, su invocación se puede hacer desde diversos proyectos implementados en distintas tecnologías. Las API proporcionan facilidad al desarrollador para implementar funcionalidades ya existentes, ahorrándole el proceso de creación de algo ya existente. Adicionalmente se puede concebir una API como un contrato, con una documentación que representa un acuerdo entre las partes, es decir, que si la parte solicitante invoca a la API con una estructura definida, esta condicionara la respuesta de la API, de esta manera por ejemplo se puede tener una integración de varias aplicaciones de distintas compañías para el intercambio de información.

1. ¿Qué es un servicio REST y Cuál es su diferencia con un SOAP?

El servicio REST (Representational State Transfer) es una interfaz para conectar varios sistemas, está basado en el uso del protocolo HTTP y define los principios de cómo se dará la comunicación entre 2 equipos. Usualmente se manejan los datos en formato JSON o XML. Su principal diferencia con un servicio SOAP radica en la simplicidad ya que SOAP al ser una alternativa más antigua, está basada en un modelo de capas, lo cual la hace más pesada, difícil de implementar y posee la limitante que solo trabaja con XML. Una ventaja adicional de los servicios REST es que al estar implementados basados en el uso del protocolo HTTP se pueden manipular los objetos a partir de la URL.