**Nombre completo: \_\_Edgar Bernardo Alfaro Medina\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**1.- Enuncie el concepto de Base de datos relacional. De algunos ejemplos de BD relacionales.** Valor 1 punto.

**Es una colección de datos organizados y almacenados en tablas estructuradas, donde las relaciones entre los datos son a través de llaves primarias y llaves foráneas.**

**Ejemplos de BD relacionales: Oracle Database, MySQL, PostgreSQL, Microsoft SQL Server.**

**2.- La siguiente sentencia obtiene la lista de estudiantes que en el año 2013 obtuvieron el mejor promedio, ¿Qué error marcará al ejecutarse? Justifique su respuesta.** Valor 1 punto.

|  |
| --- |
| **Justificación: Primero hay un error de sintaxis en el select E,APELLIDO -> Debería llevar . E.**  **El error ORA-00937 ocurre cuando se usa una función de agregación (como SUM, COUNT, AVG, etc.) sin agrupar correctamente las columnas que no forman parte de la agregación.** |

**SELECT E.NOMBRE,**

**E,APELLIDO,**

**E.ANIO,**

**SUM(E.PROMEDIO)**

**FROM ESTUDIANTES E**

**WHERE E.ANIO = 2013**

**AND E.PROMEDIO > 9;**

1. **Marcará un error ORA-00937. La función no es de grupo único.**
2. **Marcará un error porque los mejores promedios son los que obtuvieron 10 en el año 2013**
3. **Marcará un error ORA-00054 porque la tabla está bloqueada por otro proceso que la ocupa al mismo tiempo.**

**3.- De la tabla TA\_CLIENTES, deseo borrar un registro que está ligado a otra tabla llamada TA\_ORDENES por medio de una llave foránea. ¿Qué sentencia ocuparías para eliminarla? Justifique su respuesta.** Valor 1 punto.

1. **Delete from TA\_CLIENTES where ID = 1500;**
2. **Delete from TA\_CLIENTES where ID = 1500;**

**COMMIT;**

1. **Delete from TA\_ORDENES where ID = 1500;**

**Delete from TA\_CLIENTES where ID = 1500;**

**COMMIT;**

**Justificación: Cuando tienes dos tablas relacionadas mediante una llave foránea, no se puede eliminar directamente un registro de la tabla padre (TA\_CLIENTES) si existen registros relacionados en la tabla hija (TA\_ORDENES). Si intentas hacerlo, la base de datos generará un error de restricción referencial, impidiendo la eliminación para mantener la integridad de los datos.**

1. **Delete from TA\_CLIENTES where ID = 1500 ON delete cascade;**

**COMMIT;**

**4.- El proceso que obtiene los nombres de los pacientes tarda 10 minutos en obtener la lista, ¿Cómo optimizaría está consulta?.** Valor 2 puntos.

SELECT \* FROM PACIENTES;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Seleccionar unicamente la columna que se requiere, si solo se necesitan los nombres de los pacientes la consulta podría quedar así : SELECT NOMBRE FROM PACIENTES;

Esto reduce la cantidad de datos transferidos y procesados, acelerando la consulta.

**5.- Indique la estructura que debe tener un bloque de excepciones en PL/SQL y declare una excepcion llamada Proceso\_Incorrecto.** Valor 1 punto.

**DECLARE**

**Proceso\_Incorrecto EXCEPTION;**

**BEGIN**

**-- Código principal del bloque PL/SQL**

**-- Suponemos que ocurre una condición que dispara la excepción**

**IF alguna\_condicion THEN**

**RAISE Proceso\_Incorrecto; -- Dispara la excepción personalizada**

**END IF;**

**EXCEPTION**

**WHEN Proceso\_Incorrecto THEN**

**-- Código para manejar la excepción Proceso\_Incorrecto**

**DBMS\_OUTPUT.PUT\_LINE('Ha ocurrido un error: Proceso Incorrecto.');**

**WHEN OTHERS THEN**

**-- Código para manejar cualquier otra excepción no capturada**

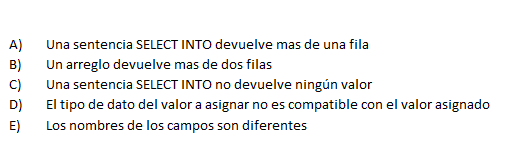
**DBMS\_OUTPUT.PUT\_LINE('Ha ocurrido un error inesperado.');**

**END;**

**6.- Relaciona ambas columnas de acuerdo a la definición de cada excepcion .** Valor 1 punto.

|  |  |
| --- | --- |
| EXCEPCION | DESCRIPCION |

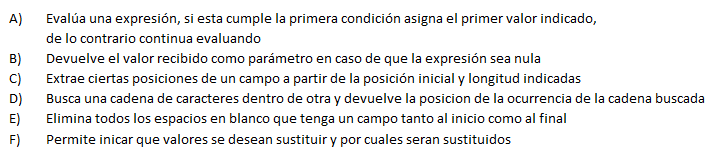
|  |  |
| --- | --- |
| NO\_DATA\_FOUND | C) |
| ROWTYPE\_MISMATCH | D), E) |
| TOO\_MANY\_ROWS | A |

****

**7.- Relaciona ambas columnas de acuerdo a la descripción que corresponde a cada función.**

Valor 1 punto.

|  |  |
| --- | --- |
| FUNCIÓN | DESCRIPCIÓN |
| TRIM | E |
| INSTR | D |
| SUBSTR | C |
| REPLACE | F |
| NVL | B |
| DECODE | A |

****

**8.- Imaginé que un médico especialista en pediatría le pide diseñar un sistema de administración de pacientes, el médico le indica que su necesidad principal es tener a la mano los datos generales de un paciente como: nombre completo, edad, sexo, dirección, medios de contacto; entre otros. Además le pide agregar la gestión de citas que hará su asistente/secretaria, las citas están clasificadas de acuerdo al tipo de paciente, por ejemplo: clientes VIP, Premium y normales, los pacientes VIP tendrán preferencia sobre el resto de los otros pacientes y es importante conocer la fecha y la hora de la cita de cada uno de sus pacientes, es importante mencionar que los pacientes pueden cancelar ó modificar una cita con 1 día de anticipación. Dibuje un modelo E-R según su experiencia que resuelve el problema (No utilice la nomenclatura de Peter Chen).** Valor 3 puntos

**Entidades y Atributos**

1. **Paciente:**

* ID\_Paciente (llave primaria)
* Nombre
* Ap Paterno
* Ap Materno
* Edad
* Sexo
* Dirección
* Telefono
* Estado

1. **Cita:**

* ID\_Cita (llave primaria)
* Fecha
* Hora
* Tipo\_Paciente (VIP, Premium, Normal)
* Estado (Programada, Cancelada, Modificada)
* **Tipo\_Paciente:**
* Tipo\_ID (llave primaria)
* Descripción (VIP, Premium, Normal)

**Relaciones**

1. **Gestiona**

* Descripción: Relaciona a un paciente con una cita.
* Atributos:
  + ID\_Cita (llave foránea)
  + ID\_Paciente (llave foránea)
  + Tipo\_Paciente (para clasificar el tipo de paciente en la cita)

1. **Cita\_Paciente**

* Descripción: Un paciente puede tener múltiples citas y cada cita pertenece a un paciente específico.

**Diagrama E-R**

1. **Paciente** (Entidad Principal)

* Relación: **Gestiona** (1 con **Cita**)
  + Un paciente puede tener varias citas, pero una cita está asociada a un solo paciente.

1. **Cita** (Entidad de Relación)

* Relación: **Gestiona** (N:1 con **Paciente**)
  + Cada cita pertenece a un paciente.

1. **Tipo\_Paciente** (Entidad de Clasificación)

* Relación: No tiene una relación directa con **Paciente**, pero se utiliza para clasificar los pacientes en la entidad **Cita**.
  + La **Cita** tiene un atributo **Tipo\_Paciente** que define si es VIP, Premium o Normal.

**Representación en el Diagrama E-R**

* **Paciente** con los atributos: ID\_Paciente, Nombre, Ap Paterno, Ap Materno, Edad, Sexo, Dirección, Telefono, Estado.
* **Cita** con los atributos: ID\_Cita, Fecha, Hora, Tipo\_Paciente, Estado.
* **Tipo\_Paciente** con los atributos: Tipo\_ID, Descripción.

Relaciones:

**Gestiona**:

**Paciente** (1) Gestiona (N:1) **Cita**

La relación **Gestiona** asocia un paciente con una o más citas y permite clasificar cada cita con un tipo de paciente.

En resumen, el modelo E-R incluye dos entidades principales (**Paciente** y **Cita**), una entidad de clasificación (**Tipo\_Paciente**), y una relación (**Gestiona**) para vincular pacientes con sus citas. La relación **Gestiona** permite la administración de las citas asociadas a cada paciente, así como el tipo de paciente (VIP, Premium, Normal), lo que permite priorizar citas según el tipo de paciente.

**9.- De acuerdo al diseño propuesto, escriba la sentencia SQL correcta para realizar la búsqueda de un paciente por nombre, apellido paterno y apellido materno.** Valor 1 punto.

SELECT \*

FROM Paciente

WHERE Nombre = 'Juan'

AND Ap\_Paterno = 'Pérez'

AND Ap\_Materno = 'Gómez';

**10.- ¿Cómo obtendría el número de pacientes activos en la base de datos? Escriba la sentencia SQL correcta.** Valor 1 punto.

SELECT COUNT(\*) AS Numero\_Pacientes\_Activos

FROM Paciente

WHERE Estado = 'Activo';

**11.- ¿Cómo obtendría la lista de pacientes que han asistido más de 2 veces en el mes de Julio? Escriba la sentencia SQL correcta.** Valor 1 punto.

SELECT P.ID\_Paciente, P.Nombre, P.Ap\_Paterno, P.Ap\_MAterno

FROM Cita C

INNER JOIN Paciente P ON C.ID\_Paciente = P.ID\_Paciente

WHERE EXTRACT(MONTH FROM C.Fecha) = 7

GROUP BY P.ID\_Paciente, P.Nombre, P.Ap\_Paterno, P.Ap\_MAterno

HAVING COUNT(\*) > 2;

**12.- Complete el siguiente procedimiento que inserta un paciente nuevo. No olvide el manejo de transacciones.** Valor 3 puntos.

CREATE OR REPLACE PROCEDURE MEDICOS.SPPACIENTES (

PANOMBRE IN \_\_VARCHAR2\_\_\_\_\_,

PAAPPATERNO IN \_\_VARCHAR2\_\_\_\_\_,

PAAPMATERNO IN \_\_ VARCHAR2 \_\_\_,

PASEXO IN \_\_\_VARCHAR2\_\_\_\_\_,

PADIRECCION IN \_\_\_VARCHAR2\_\_\_\_,

PACORREO IN \_\_\_ VARCHAR2\_\_\_,

PATELEFONO IN \_\_\_ VARCHAR2\_\_\_\_\_\_,

IS

/\*\* Área disponible para declaraciones \*\*/

BEGIN

/\*\* Área disponible para código\*\*/

INSERT INTO Paciente (

Nombre,

Ap\_Paterno,

Ap\_Materno,

Sexo,

Direccion,

Correo,

Telefono

)

VALUES (

PANOMBRE,

PAAPPATERNO,

PAAPMATERNO,

PASEXO,

PADIRECCION,

PACORREO,

PATELEFONO );

COMMIT;

EXCEPTION

WHEN OTHERS THEN

ROLLBACK;

END SPPACIENTES;

**13.- Complete la siguiente función que obtiene un valor de retorno numérico cuando el paciente se encuentra dado de baja de la base de datos (Escriba la sentencia correcta sobre la línea).** Valor 2 puntos.

CREATE OR REPLACE FUNCTION MEDICOS.FNVALIDAR (

PAPACIENTE \_\_\_\_\_**IN NUMBER**\_\_\_\_\_\_\_\_\_\_ )

RETURN NUMBER

IS\_\_\_

/\*\* Declaraciones \*\*/

\_\_\_\_\_\_\_\_**VLEXISTE NUMBER;**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

BEGIN

SELECT COUNT(1)

INTO VLEXISTE

FROM SADE.PACIENTES TS

WHERE **TS.ID\_PACIENTE = PAPACIENTE**

AND TS.FNACTIVO = 0;

IF VLEXISTE = 1 THEN

VLEXISTE := 1;

ELSE

\_**VLEXISTE := 0**;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

END IF;

EXCEPTION

WHEN OTHERS THEN

\_\_\_\_**RETURN –1;\_**\_\_\_\_  **-- -1 puede ser un código para indicar error en la ejecución**

END;

/

**14.- La función que fue creada para obtener sí un paciente está dado de baja no cuenta con permisos de ejecución, ¿Cuál es la sentencia correcta para asignarle permisos de ejecución al usuario USRMEDICOS?** Valor 1 puntos.

**a)** GRANT EXECUTE ON MEDICOS.FNVALIDAR TO USRMEDICOS;

**b)** REVOKE EXECUTE ON MEDICOS.FNVALIDAR TO USRMEDICOS;

**c)** GRANT EXECUTE ON MEDICOS.FNVALIDAR TO MEDICOS;

**d)** GRANT EXECUTE ON MEDICOS.SPPACIENTES TO USRMEDICOS;