

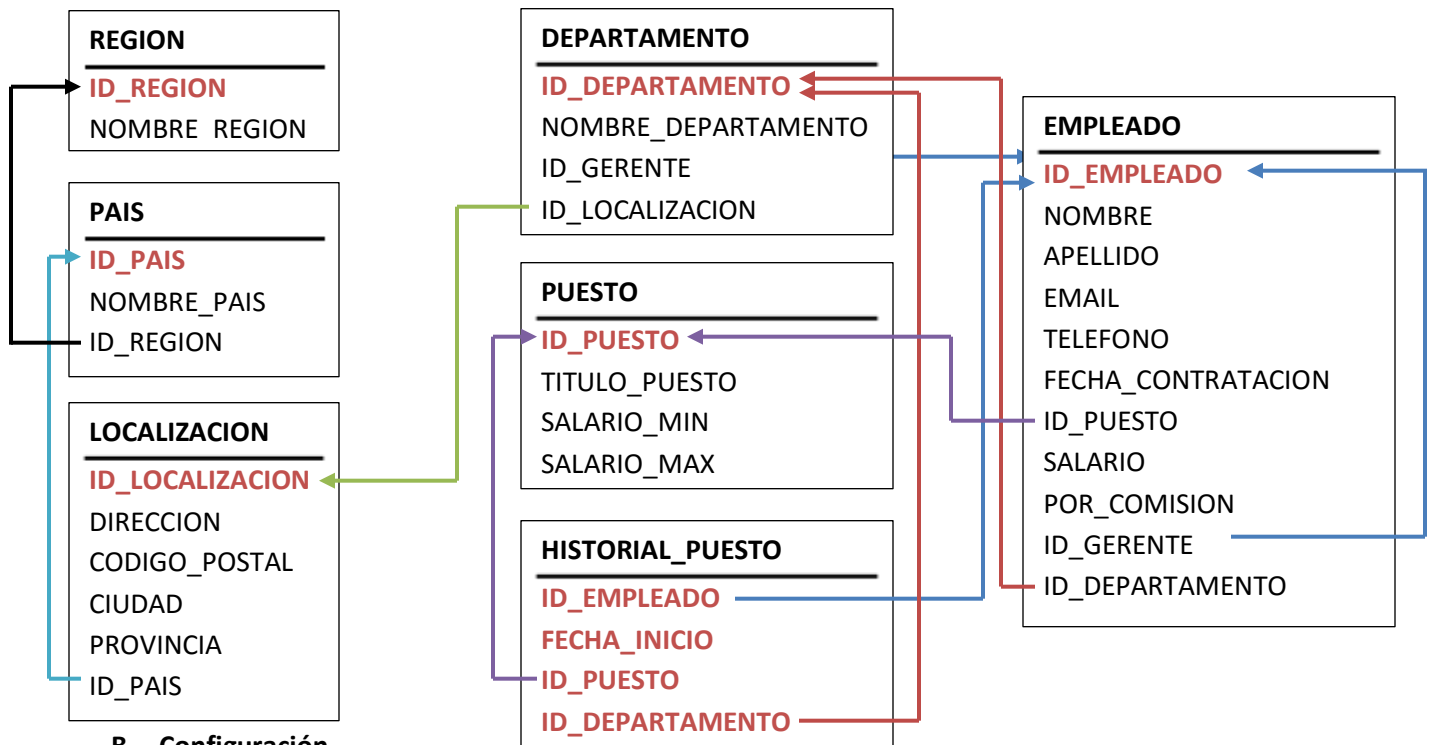
Tarea Corta #1 ***(Grupal)***

A. Instrucciones:

- B. La tarea debe entregarse al profesor por medio del TEC Digital en la fecha y hora establecida
- C. La tarea debe tener el siguiente nombre: GrupoNo#_BDII_**Tarea Corta#1**
- D. La tarea debe contener lo siguiente:
- E. Todo el código fuente necesario para ejecutar la tarea debidamente documentado y escrito en ORACLE PL/SQL.
- F. De ser necesario, la tarea deberá ser defendida ante el profesor si este lo considera necesario, de tal manera que todos los estudiantes deben poder explicar la solución satisfactoriamente.
- G. La revisión de la tarea la realizará el profesor con cada grupo donde se deben probar todos o parte de los entregables.

Tarea Corta No.1

Esquema de la Base de Datos



B. Configuración

1. Normalización: verifique si la base de datos cumple con las tres primeras formas normales y de no ser así donde se viola alguna de las formas normales o en que eventuales casos se podrían presentar problemas que atenten con la normalización.
2. Configure 2 tablespaces, uno para datos y otro para los índices con las siguientes características
 - a. No puede ser auto extendido
 - b. Debe ser regular con un bloque de 8K
 - c. Debe ser permanente
 - d. Use el estándar <GRUPO#>_(TBL|IDX) para los nombres, por ejemplo, GRUPO1_TBL y GRUPO1_IDX.
3. Cree un nuevo usuario (OWNER) para asociar el esquema y cree todos los objetos con sus respectivos índices a este nuevo esquema. Use el estándar <INICIAL NOMBRE> <APELLIDO> para el usuario, por ejemplo MCorea.
4. Crear a todos los usuarios de su grupo los cuales tendrán diferentes roles con permisos para poder acceder los diferentes objetos.
5. Crear al menos tres roles: uno con permiso para ejecutar los procedimientos y triggers (ROL_EJECUTA); otro con accesos limitados sobre algunos de los objetos (ROL_NOVATO) y un tercer rol con permisos ilimitados (ROL_SUPERIOR).
6. Adjunte la evidencia necesaria que muestre el cumplimiento del esquema de seguridad.

C. Genere el código PL/SQL para las siguientes consultas

1. Despliegue todos los años donde más de 15 empleados hayan entrado a la compañía.
2. Despliegue todos los departamentos donde haya al menos un empleado que reciba comisiones.
3. Despliegue todos los empleados que hayan cambiado de puesto en el último año.
4. Muestre por nombre y apellido de empleado la cantidad de días que lleva haciendo el trabajo actual.
5. Despliegue nombre y apellido de empleado y el salario promedio para todos los empleados que reciben comisiones.
6. Muestre todos los nombres de empleados que llevan más de un año trabajando y su salario sea más de \$10000.
7. Despliegue todos los trabajos para los cuales se contrató personal en el último año.
8. Despliegue toda la localización geográfica, país, ciudad y departamento, para aquellos departamentos que tienen más de 10 empleados.
9. Despliegue todas las personas que tienen al menos 10 personas a cargo.

D. Genere el código PL/SQL para lo siguiente

1. Incremente el salario de los empleados en un 10%, si han trabajado en la empresa más de 10 años, un 5%, más de 5 años, 2%, en cualquier otro caso.
2. Muestre usando dbms_output.put_line cuál es el años en que más empleados fueron contratados en la compañía y para ese año cuantos fueron contratados por mes.
3. Despliegue usando dbms_output.put_line todos los nombres y apellido de empleados que fueron los primeros en ocupar un puesto, junto con el puesto y la fecha en que iniciaron.
4. Cree una función que tome el identificador del departamento y retorne el nombre del manager del departamento.
5. Cree una función que tome el identificador del empleado y retorne cuantos puestos ha tenido en la compañía.
6. Cree un procedimiento que tome el identificador un departamento y cambie al manager por el empleado con mayor antigüedad dentro del departamento.
7. Cree un procedimiento que tome el identificador del departamento y retorne todos los empleados con nombre y apellido ordenados en forma descendente según su salario.
8. Cree un trigger que asegure que el salario no puede ser negativo.
9. Cree un trigger que inserte en la tabla (AUDITORIA) cada vez que se actualice el salario de un empleado. La tabla debe contener la fecha actual con la hora, el nombre del usuario que realizo el cambio, el monto anterior y el monto actual

Mucho éxito!