**1º DAM/DAW BASES DE DATOS**

1. **Una empresa almacena los datos de sus empleados en una tabla llamada “empleados”. Crear la tabla e insertar los registros utilizando este script.**

create table empleados(

documento char(8),

nombre varchar2(20),

apellido varchar2(20),

sueldo number(6,2),

fechaingreso date

);

insert into empleados values('22222222','Juan','Perez',300,'10/10/1980');

insert into empleados values('22333333','Luis','Lopez',300,'12/05/1998');

insert into empleados values('22444444','Marta','Perez',500,'25/08/1990');

insert into empleados values('22555555','Susana','Garcia',400,'05/05/2000');

insert into empleados values('22666666','Jose Maria','Morales',400,'24/10/2005');

**Crear un procedimiento llamado "pa\_empleados\_aumentarsueldo". Debe incrementar el sueldo de los empleados con cierta cantidad de años en la empresa (parámetro "ayear" de tipo numérico) en un porcentaje (parámetro "aporcentaje" de tipo numerico); es decir, recibe 2 parámetros.**

**Ejecutar el procedimiento de modo que se aumente el sueldo a los empleados que tengan más de 10 años en un 20%.**

**Ejecutar el procedimiento para los que tengan más de ocho años en la empresa y les aumente un 10%**

**Ejecutar el procedimiento "pa\_empleados\_aumentarsueldo" sin parámetros’ ¿es posible? ¿por qué?**

**Crear un procedimiento almacenado llamado "pa\_empleados\_ingresar" que inserte un empleado en la tabla "empleados", debe recibir valor para el documento, el nombre, apellido y almacenar valores nulos en los campos "sueldo" y "fechaingreso"**

**Reemplazar el procedimiento almacenado llamado "pa\_empleados\_ingresar" para que ingrese un empleado en la tabla "empleados", debe recibir valor para el documento (con valor por defecto nulo) y fechaingreso (con la fecha actual como valor por defecto), los demás campos se llenan con valor nulo**

**Ejecutar el procedimiento creado anteriormente enviándole valores para los 2 parámetros y verifique si se ha ingresado en "empleados" un nuevo registro**

**Ejecutar el procedimiento creado anteriormente enviando solamente la fecha de ingreso y comprobar el resultado**

**Crear un procedimiento que reciba un documento y elimine de la tabla "empleados" el empleado que coincida con dicho documento**

**Eliminar un empleado empleando el procedimiento del punto anterior**

1. **Dado el siguiente script:**

create table notas(

nombre varchar2(30),

nota number(4,2)

);

insert into notas values('Acosta Ana', 6.7);

insert into notas values('Bustos Brenda', 9.5);

insert into notas values('Caseros Carlos', 3.7);

insert into notas values('Dominguez Daniel', 2);

insert into notas values('Fuentes Federico', 8);

insert into notas values('Gonzalez Gaston', 7);

insert into notas values('Juarez Juana', 4);

insert into notas values('Lopez Luisa',5.3);

**Crea la función “f\_condicion” que recibe una nota y retorna una cadena de caracteres indicando si aprueba o no:**

1. **Un negocio almacena los datos de sus productos en una tabla denominada "productos". Dicha tabla contiene el código de producto, el precio, el stock mínimo que se necesita (cantidad mínima requerida antes de comprar más) y el stock actual (cantidad disponible en depósito). Si el stock actual es cero, es urgente reponer tal producto; menor al stock mínimo requerido, es necesario reponer tal producto; si el stock actual es igual o supera el stock mínimo, está en estado normal.**

**Utiliza el script:**

create table productos(

codigo number(5),

precio number(6,2),

stockminimo number(4),

stockactual number(4)

);

insert into productos values(100,10,100,200);

insert into productos values(102,15,200,500);

insert into productos values(130,8,100,0);

insert into productos values(240,23,100,20);

insert into productos values(356,12,100,250);

insert into productos values(360,7,100,100);

insert into productos values(400,18,150,100);

**Crear una función que reciba dos valores numéricos correspondientes a ambos stocks. Debe comparar ambos stocks y retornar una cadena de caracteres indicando el estado de cada producto, si stock actual es:**

**- cero: "faltante",**

**- menor al stock mínimo: "reponer",**

**- igual o superior al stock mínimo: "normal".**

**Realizar un "select" mostrando el código del producto, ambos stocks y, empleando la función creada anteriormente, una columna que muestre el estado del producto**

**Realizar una función similar a las anteriores, pero esta vez, si el estado es "reponer" o "faltante", debe especificar la cantidad necesaria (valor necesario para llegar al stock mínimo)**

**Realizar un "select" mostrando el código del producto, ambos stocks y, empleando la función creada anteriormente, una columna que muestre el estado del producto**

1. **Una clínica almacena la información de sus pacientes en una tabla denominada “pacientes”.**

create table pacientes(

documento char(8),

nombre varchar2(30),

edad number(2),

sexo char(1)

);

insert into pacientes values('11111111','Acosta Ana',40,'f');

insert into pacientes values('22222222','Bustos Betina',35,'f');

insert into pacientes values('33333333','Caseres Carlos',18,'m');

insert into pacientes values('44444444','Dominguez Diego',6,'m');

insert into pacientes values('15555555','Fuentes Fabiana',55,'f');

insert into pacientes values('26666666','Gomez Gaston',38,'m');

insert into pacientes values('37777777','Irala Ines',16,'f');

insert into pacientes values('38888888','Juarez Julieta',17,'f');

insert into pacientes values('40000000','Lopez Lucas',3,'m');

**Realizar una función que reciba la edad del paciente y retorne la cadena "menor" o "mayor" según sea menor a 18 años o no**

**Realizar una función que reciba el carácter correspondiente al sexo del paciente y retorne la cadena "femenino" o "masculino"**

**Realizar un "select" mostrando el nombre del paciente y empleando las funciones de los puntos 3 y 4, dos columnas que indiquen si es mayor o menor de edad y el sexo.**

1. **Crear la siguiente tabla e insertar los registros utilizando los siguientes scripts.**

create table empleados (

documento char(8),

nombre varchar(30),

fechanacimiento date,

hijos number(2),

sueldo number(6,2),

sexo char(1)

);

insert into empleados values('20000000','Acosta Ana','10/05/1968',0,800,'f');

insert into empleados values('21111111','Bustos Bernardo','09/07/1970',2,550,'m');

insert into empleados values('22222222','Caseros Carlos','15/10/1971',3,500,'m');

insert into empleados values('23333333','Fuentes Fabiana','25/08/1972',0,500,'f');

insert into empleados values('24444444','Gomez Gaston','28/03/1979',1,850,'m');

insert into empleados values('25555555','Juarez Javier','18/08/1981',2,600,'m');

insert into empleados values('26666666','Lopez Luis','17/09/1978',4,690,'m');

insert into empleados values('27777777','Morales Marta','22/08/1975',2,480,'f');

insert into empleados values('28888888','Norberto Nores','11/08/1973',3,460,'m');

insert into empleados values('29999999','Oscar Oviedo','19/07/1976',0,700,'m');

**La empresa tiene por política festejar los cumpleaños de sus empleados cada mes, si es de sexo femenino se le regala un ramo de flores, sino, una agenda. Realice un "select" mostrando el nombre del empleado, el día del cumpleaños y una columna extra que muestre "FLORES" o "AGENDA" según el sexo del empleado, de todos los empleados que cumplen años en el mes de agosto y ordene por día.**

1. **Un profesor guarda los promedios de sus alumnos de un curso en una tabla llamada "alumnos". Crear la tabla e insertar los registros que se indican, utilizando los siguientes scripts.**

create table alumnos(

legajo char(5) not null,

nombre varchar2(30),

promedio number(4,2)

);

insert into alumnos values(3456,'Perez Luis',8.5);

insert into alumnos values(3556,'Garcia Ana',7.0);

insert into alumnos values(3656,'Ludueña Juan',9.6);

insert into alumnos values(2756,'Moreno Gabriela',4.8);

insert into alumnos values(4856,'Morales Hugo',3.2);

insert into alumnos values(7856,'Gomez Susana',6.4);

**Si el alumno tiene un promedio menor a 4, muestre un mensaje "suspenso", si el promedio es mayor o igual a 4 y menor a 7, muestre "aprobado", si el promedio es mayor o igual a 7, muestre "notable", usando "case" (recuerde que "case" toma valores puntuales, emplee "trunc")**

1. **Crear una tabla denominada "alumnosCondicion"**

create table alumnosCondicion(

legajo char(5),

notafinal number(4,2),

condicion varchar2(15)

);

**Crear un procedimiento llamado "pa\_CargarCondicion" que guarde en la tabla "alumnosCondicion" el legajo de cada alumno, el promedio de sus notas y la condición (suspenso de 0 a 3, aprobado de 4 a 6 o notable mayor de 6)**

**Ejecutar el procedimiento "pa\_cargarCondicion" y recuperar todos los datos de la tabla "alumnoscondicion"**

1. **Una empresa almacena los datos de sus empleados en una tabla denominada "empleados".**

drop table empleados;

create table empleados(

nombre varchar2(40),

sueldo number(6,2)

);

insert into empleados values('Acosta Ana',550);

insert into empleados values('Bustos Bernardo',850);

insert into empleados values('Caseros Carolina',900);

insert into empleados values('Dominguez Daniel',490);

insert into empleados values('Fuentes Fabiola',820);

insert into empleados values('Gomez Gaston',740);

insert into empleados values('Huerta Hernan',1050);

**Mostar la suma total de todos los sueldos**

**Se necesita incrementar los sueldos en forma proporcional, en un 10% cada vez y controlar que la suma total de sueldos no sea menor a 7000, si lo es, el bucle debe continuar y volver a incrementar los sueldos, en caso de superarlo, se saldrá del ciclo repetitivo; es decir, este bucle continuará el incremento de sueldos hasta que la suma de los mismos llegue o supere los 7000.**

**Verificar que los sueldos han sido incrementados y la suma de todos los sueldos es superior a 7000**

**Mostrar el sueldo máximo**

**Se necesita incrementar los sueldos en forma proporcional, en un 5% cada vez y controlar que el sueldo máximo alcance o supere los 1600, al llegar o superarlo, el bucle debe finalizar. Incluya una variable contador que cuente cuántas veces se repite el bucle.**

**Verificar que los sueldos han sido incrementados y el sueldo máximo es igual o superior a 1600**

**Mostrar el sueldo mínimo realizando un "select"**

**Se necesita incrementar los sueldos en forma proporcional, en un 10% cada vez y controlar que el sueldo mínimo no supere los 900. Emplee la sintaxis "if CONDICION then exit"**

1. **Dentro del ciclo debe haber una estructura condicional que controle que el número sea par y si lo es, lo imprima por pantalla.**
2. **Con la estructura repetitiva "for... loop" muestre el sumatorio del número 5; la suma de todos los números del 1 al 5. Al finalizar el ciclo debe mostrarse por pantalla el sumatorio de 5 (15)**
3. **Cree una función que reciba un valor entero y retorne el factorial de tal número; el factorial se obtiene multiplicando el valor que recibe por el anterior hasta llegar a multiplicarlo por uno)**

**Llame a la función creada anteriormente y obtenga el factorial de 5 y de 4 (120 y 24)**

1. **Crear un procedimiento que reciba dos parámetros numéricos; se debe mostrar la tabla de multiplicar del número enviado como primer argumento, desde el 1 hasta el número enviado como segundo argumento. Emplee "for"**

**Execute el procedimiento creado anteriormente enviándole los valores necesarios para que muestre la tabla del 6 hasta el 20**

**Execute el procedimiento creado anteriormente enviándole los valores necesarios para que muestre la tabla del 9 hasta el 10**