

PROGRAMACIÓN DE BASE DE DATOS

EVALUACIÓN I

Competencia asociada:

Desarrollar back end de aplicaciones utilizando estándares de la industria.

Unidad de Competencia:

Opera sobre una base de datos utilizando lenguaje PL/SQL, permitiendo organizar la lógica de operación de los datos y controlar los errores.

Consideraciones importantes

1. La evaluación tiene solo un intento permitido.
2. Usted posee un periodo en envío limitado para poder realizar el encargo. Luego de ello se entrega la retroalimentación de su código
3. La copia o plagio de algún encargo y/o evaluación será sancionada con nota 1.

Instrucciones

1. De acuerdo a los requerimientos, debe realizar el encargo con SQL Developer.
2. El encargo debe ser subido en el ambiente de aprendizaje en el recurso Entrega de Evaluación, en donde debe entregar el archivo .SQL con comentarios de solución.
3. En los comentarios deben indicar los pasos que pensó para encontrar la solución en el recurso entrega de evaluación.

Utilizando el esquema HR realice los siguientes requerimientos:

Requerimiento 1

Se requiere crear un bloque PL/SQL anónimo que pueda visualizar el total de empleados que trabajan en el departamento de finanzas. Las especificaciones que se deben considerar para la creación del bloque son:

- Definir la variable **v_total_dep** con su tipo de dato numérica con un largo de 3 e inicializada en cero.
- Definir la variable **v_dep** con su tipo de dato de carácter con un largo de 40.
- Obtener el nombre del departamento y el total de empleados que trabajen en el departamento de finanzas y almacenar los valores en las variables definidas.
- Se debe mostrar la información en el siguiente formato:

```
En el departamento Finance trabajan 6 empleados
```

Requerimiento 2

Usted debe categorizar a los empleados con identificación 100 al 130, clasificando su salario. Para ello es necesario crear un bloque PL/SQL que muestre la identificación del empleado, su salario, seguido del mensaje que corresponda según el valor del sueldo que posee:

- Si el salario es mayor a 10000 el mensaje debe ser **‘Es un buen sueldo’**.
- Si el salario está entre 10000 y los 9000 el mensaje debe ser **‘Es un salario normal’**.
- Si el salario es menor a 9000 el mensaje debe ser **‘Es un sueldo que se debe aumentar’**.

```
El salario actual del empleado 100 es de 24000. Es un buen sueldo
El salario actual del empleado 101 es de 17000. Es un buen sueldo
El salario actual del empleado 102 es de 17000. Es un buen sueldo
El salario actual del empleado 103 es de 9000. Es un salario normal
El salario actual del empleado 104 es de 6000. Es un sueldo que se debe aumentar
El salario actual del empleado 105 es de 4800. Es un sueldo que se debe aumentar
El salario actual del empleado 106 es de 4800. Es un sueldo que se debe aumentar
El salario actual del empleado 107 es de 4200. Es un sueldo que se debe aumentar
El salario actual del empleado 108 es de 12008. Es un buen sueldo
El salario actual del empleado 109 es de 9000. Es un salario normal
El salario actual del empleado 110 es de 8200. Es un sueldo que se debe aumentar
El salario actual del empleado 111 es de 7700. Es un sueldo que se debe aumentar
El salario actual del empleado 112 es de 7800. Es un sueldo que se debe aumentar
El salario actual del empleado 113 es de 6900. Es un sueldo que se debe aumentar
El salario actual del empleado 114 es de 11000. Es un buen sueldo
El salario actual del empleado 115 es de 3100. Es un sueldo que se debe aumentar
El salario actual del empleado 116 es de 2900. Es un sueldo que se debe aumentar
El salario actual del empleado 117 es de 2800. Es un sueldo que se debe aumentar
El salario actual del empleado 118 es de 2600. Es un sueldo que se debe aumentar
El salario actual del empleado 119 es de 2500. Es un sueldo que se debe aumentar
```

Requerimiento 3

Usted deberá identificar los problemas que se encuentran en el siguiente código. Entregando la solución óptima y comentar los errores detectados:

```
DECLARE
    v_msj VARCHAR2(10);
    v_prom NUMBER;
    v_nombre NUMBER(20);
    v_sum NUMBER;

BEGIN

    SELECT round(avg(salary),0),department_name,sum(salary)
    INTO v_prom,v_nombre,v_sum ,v_count
    FROM employees join departments
    USING (department_id)
    GROUP BY department_name,department_id
    HAVING department_id = &Ingrese_ID_departamento;

    IF sql%found THEN
        DBMS_OUTPUT.PUT_LINE('Se seleccionaron: '|| sql%rowcount || ' Filas');
    ELSE
        DBMS_OUTPUT.PUT_LINE('No se selcciono nada');
    END IF;

    v_msj := IF WHEN v_prom<5000 THEN 'Excelente'
               WHEN v_prom<3000 THEN 'Bueno'
               WHEN v_prom<1000 THEN 'Inaceptable '
               ELSE '-'
               END;

    DBMS_OUTPUT.PUT_LINE('Su valoracion es : '||v_msj||' ,el valor de promedio es: '
                        ||(v_prom,'$999g999')
                        ||' y el nombre del departamento es: '
                        ||v_nombre);

END;
```

Requerimiento 4

Se requiere crear un bloque PL/SQL anónimo que permita cubrir las siguientes especificaciones.

- Mostrar el total de empleados que trabajen en el departamento 50.
- Se requiere saber el empleado con el salario más bajo.
- Mostrar el valor promedio de las comisiones que se calculan a los empleados. Se debe considerar que el valor de la comisión es salario * **porcentaje_comision** y que para obtener el promedio se deben considerar TODOS los empleados posean o no porcentaje de comisión.
- Mostrar el ID, nombre, jefe y ubicación del departamento con más empleados en la organización:
- Se debe mostrar la información en el siguiente formato:

```
En el departamento Shipping trabajan 45 empleados.
TJ Olson es el empleado con el salario más bajo.
El promedio de las comisiones es $689
La información del departamento con mayor cantidad de empleados es la siguiente:
Identificación: 50
Nombre: Shipping
Jefe: 121
Ubicación: 1500
```

Criterios de evaluación

Escala de apreciación

Indicadores de logro	COMPLETAMENTE LOGRADO	MEDIANAMENTE LOGRADO	NO LOGRADO
	2 PUNTOS	1 PUNTO	0 PUNTO
1. Construye los Bloques Anónimos, de acuerdo a los requerimientos.			
2. Aplica adecuadamente en la declaración de variables los tipos de datos, de acuerdo a los requerimientos entregados en cada caso.			
3. Aplica y construye adecuadamente la estructura de control en PL/SQL, de acuerdo a los requerimientos.			
4. Aplica y construye adecuadamente las sentencias IF o CASE, dependiendo el caso entregado.			
5. Comenta los códigos desarrollados paso a paso.			
6. Detecta y comenta el requerimiento 3, con los problemas y soluciones correspondientes.			
7. Muestra los resultados con la misma estructura de ejemplo en los requerimientos.			

AUTOEVALUACIÓN

Estimado estudiante:

La autoevaluación es la valoración y/o reflexión que usted realiza sobre su proceso de aprendizaje. Este ejercicio es fundamental para identificar tanto debilidades como fortalezas de la evaluación realizada.

Para realizar la autoevaluación usted deberá contestar las siguientes preguntas, las cuales se encontrarán en la pestaña **AUTOEVALUACIÓN**, donde deberá completar cada una de ellas. *(No se deberán contestar en esta planilla)*

Conteste con toda la sinceridad posible, ya que será un insumo para la retroalimentación que realizará posteriormente el docente.

- ¿Qué hice bien?
- ¿En qué necesito ayuda?
- ¿Sobre qué quiero saber más?
- ¿Qué aprendí en esta evaluación?
- ¿Qué debo mejorar?
- ¿Qué acciones realizaré para mejorar?