

IESTP

“ANDRÉS AVELINO CÁCERES DORREGARAY”

DISEÑO Y PROGRAMACIÓN WEB



Actividad 06

CURSO : Programación de Aplicaciones Web y Móviles

TURNO : Diurno - VI

PROFESOR : Raúl Fernandez

ALUMNO : Quispe Osorio Luis Alberto

2024

DESARROLLO DE EJERCICIOS

Proyecto 01: Cálculo de Rebajas en Pensiones Universitarias

Enunciado del Proyecto

En una universidad, los estudiantes están clasificados en cuatro categorías (A, B, C, D), cada una con una pensión mensual específica. Semestralmente, la universidad aplica descuentos en las pensiones de los estudiantes a partir del segundo ciclo, basándose en el promedio ponderado del ciclo anterior. El programa debe determinar la rebaja aplicada y la nueva pensión del estudiante, dada su categoría y promedio ponderado.

Requerimientos Funcionales

1. Entrada de Datos:
 - Categoría del Estudiante: A, B, C o D.
 - Promedio Ponderado del Ciclo Anterior: Valor numérico entre 0 y 20.
 2. Procesamiento:
 - Determinar la Pensión Base: Según la categoría ingresada.
 - Calcular el Porcentaje de Descuento: Basado en el promedio ponderado según la tabla de descuentos.
 - Calcular el Monto de la Rebaja: Aplicar el porcentaje de descuento a la pensión base.
 - Determinar la Nueva Pensión: Restar la rebaja de la pensión base.
 3. Salida de Datos:
 - Monto de la Rebaja: Valor en soles.
 - Nueva Pensión Mensual: Valor en soles.
 4. Validaciones:
 - Verificar que la categoría ingresada sea válida (A, B, C, D).
 - Asegurar que el promedio esté dentro del rango 0 a 20.
-

Diagrama de Clases

plaintext

Copiar código

+-----+

| Estudiante |

```

+-----+
| - categoria: Char |
| - promedio: Float |
+-----+
| + calcularRebaja() |
| + nuevaPension() |
+-----+

```

```

+-----+
|   Pension   |
+-----+
| - base: Float   |
| - descuento: Float|
| - nuevaPension: Float |
+-----+
| + aplicarDescuento() |
+-----+

```

<<enumeration>>

DescuentoPromedio

```

+-----+
| 0-13.99: 0%   |
| 14-15.99: 10%  |
| 16-17.99: 12%  |
| 18-20: 15%    |
+-----+

```

Descripción de Clases:

- Estudiante:
 - Atributos:
 - categoria: Representa la categoría del estudiante (A, B, C, D).
 - promedio: Promedio ponderado del ciclo anterior.

- Métodos:
 - `calcularRebaja()`: Determina el porcentaje de descuento basado en el promedio.
 - `nuevaPension()`: Calcula la nueva pensión aplicando la rebaja.
 - Pension:
 - Atributos:
 - `base`: Pensión base según la categoría.
 - `descuento`: Monto de la rebaja aplicada.
 - `nuevaPension`: Pensión después de aplicar la rebaja.
 - Métodos:
 - `aplicarDescuento()`: Calcula el descuento y la nueva pensión.
 - DescuentoPromedio (Enumeración):
 - Define los rangos de promedio y sus correspondientes porcentajes de descuento.
-

Proyecto 02: Evaluación de Empleados y Cálculo de Bonificaciones

Enunciado del Proyecto

Una empresa evalúa a sus empleados basándose en dos criterios: puntualidad y rendimiento. Cada criterio otorga un puntaje específico según los minutos de tardanza y el número de observaciones. La suma de ambos puntajes determina la bonificación anual del empleado. El programa debe calcular el puntaje por puntualidad, el puntaje por rendimiento, el puntaje total y la bonificación correspondiente, dados los minutos de tardanza y el número de observaciones del empleado.

Requerimientos Funcionales

1. Entrada de Datos:
 - Minutos de Tardanza: Entero ≥ 0 .
 - Número de Observaciones: Entero ≥ 0 .
2. Procesamiento:
 - Calcular Puntaje por Puntualidad: Según la tabla de minutos de tardanza.
 - Calcular Puntaje por Rendimiento: Según la tabla de observaciones.
 - Calcular Puntaje Total: Suma de los puntajes anteriores.

- Determinar Bonificación: Basada en el puntaje total según la tabla de bonificaciones.

3. Salida de Datos:

- Puntaje por Puntualidad.
- Puntaje por Rendimiento.
- Puntaje Total.
- Monto de la Bonificación Anual.

4. Validaciones:

- Verificar que los minutos de tardanza y las observaciones sean números enteros no negativos.

Diagrama de Clases

plaintext

Copiar código

```
+-----+
| Empleado |
+-----+
| - minutosTarde: Int |
| - observaciones: Int|
+-----+
| + calcularPuntajePuntualidad() |
| + calcularPuntajeRendimiento() |
| + calcularPuntajeTotal()      |
| + calcularBonificacion()      |
+-----+

+-----+
| Puntaje |
+-----+
| - puntajePuntualidad: Int |
| - puntajeRendimiento: Int |
| - puntajeTotal: Int      |
```

```

+-----+
| + sumarPuntajes() |
+-----+

+-----+
| Bonificacion |
+-----+

| - montoPorPunto: Float |
| - totalBonificacion: Float |
+-----+
| + determinarBonificacion() |
+-----+

```

Descripción de Clases:

- Empleado:
 - Atributos:
 - minutosTarde: Minutos de tardanza.
 - observaciones: Número de observaciones.
 - Métodos:
 - calcularPuntajePuntualidad(): Asigna puntaje según minutos de tardanza.
 - calcularPuntajeRendimiento(): Asigna puntaje según observaciones.
 - calcularPuntajeTotal(): Suma de ambos puntajes.
 - calcularBonificacion(): Calcula la bonificación basada en el puntaje total.
- Puntaje:
 - Atributos:
 - puntajePuntualidad: Puntaje obtenido por puntualidad.
 - puntajeRendimiento: Puntaje obtenido por rendimiento.
 - puntajeTotal: Suma de ambos puntajes.
 - Métodos:
 - sumarPuntajes(): Calcula el puntaje total.
- Bonificacion:

- Atributos:
 - montoPorPunto: Monto de soles por punto según el rango.
 - totalBonificacion: Total de la bonificación anual.
 - Métodos:
 - determinarBonificacion(): Determina el monto por punto y calcula la bonificación total.
-

Proyecto 03: Sistema de Ventas y Descuentos en una Dulcería

Enunciado del Proyecto

Una dulcería vende diferentes tipos de chocolates a precios establecidos. Además, ofrece descuentos basados en la cantidad de chocolates adquiridos y obsequios de caramelos según el importe total de la compra. El programa debe calcular el importe de la compra, el descuento aplicado, el importe a pagar y la cantidad de caramelos de obsequio, dados el tipo de chocolate y la cantidad de unidades adquiridas.

Requerimientos Funcionales

1. Entrada de Datos:
 - Tipo de Chocolate: Primor, Dulzura, Tentación o Explosión.
 - Cantidad de Unidades Adquiridas: Entero ≥ 1 .
2. Procesamiento:
 - Determinar el Precio Unitario: Según el tipo de chocolate.
 - Calcular el Importe de la Compra: Precio unitario \times cantidad.
 - Determinar el Porcentaje de Descuento: Según la cantidad de chocolates.
 - Calcular el Monto del Descuento: Aplicar el porcentaje al importe de la compra.
 - Calcular el Importe a Pagar: Importe de la compra menos el descuento.
 - Determinar la Cantidad de Caramelos de Obsequio:
 - Si importe a pagar \geq S/. 250: 3 caramelos por chocolate.
 - Si importe a pagar $<$ S/. 250: 2 caramelos por chocolate.
3. Salida de Datos:
 - Importe de la Compra.
 - Monto del Descuento.

- Importe a Pagar.
- Cantidad de Caramelos de Obsequio.

4. Validaciones:

- Verificar que el tipo de chocolate ingresado sea válido.
- Asegurar que la cantidad de unidades sea un número entero positivo.

Diagrama de Clases

plaintext

Copiar código

```
+-----+
|  Chocolate  |
+-----+
| - tipo: String |
| - cantidad: Int |
+-----+
| + obtenerPrecioUnitario() |
| + calcularImporte()      |
| + determinarDescuento() |
| + calcularDescuento()    |
| + calcularImportePagar() |
| + determinarCaramelos() |
+-----+

+-----+
|  Descuento  |
+-----+
| - porcentaje: Float |
+-----+
| + aplicarDescuento()|
+-----+
```



```

+-----+
|  Obsequio  |
+-----+

| - caramelosPorUnidad: Int |
| - totalCaramelos: Int    |
+-----+

| + calcularCaramelos()    |
+-----+

```

Descripción de Clases:

- Chocolate:
 - Atributos:
 - tipo: Tipo de chocolate (Primor, Dulzura, Tentación, Explosión).
 - cantidad: Número de unidades adquiridas.
 - Métodos:
 - obtenerPrecioUnitario(): Retorna el precio según el tipo.
 - calcularImporte(): Calcula el importe de la compra.
 - determinarDescuento(): Determina el porcentaje de descuento según la cantidad.
 - calcularDescuento(): Calcula el monto del descuento.
 - calcularImportePagar(): Calcula el importe a pagar después del descuento.
 - determinarCaramelos(): Calcula la cantidad de caramelos de obsequio.
- Descuento:
 - Atributos:
 - porcentaje: Porcentaje de descuento aplicable.
 - Métodos:
 - aplicarDescuento(): Calcula el monto del descuento basado en el porcentaje.
- Obsequio:
 - Atributos:
 - caramelosPorUnidad: Número de caramelos por cada chocolate.
 - totalCaramelos: Total de caramelos a obsequiar.

- Métodos:
 - `calcularCaramelos()`: Determina el total de caramelos según el importe a pagar.

Proyecto 04: Sistema de Ventas y Regalos en una Tienda de Productos

Enunciado del Proyecto

Una tienda comercializa tres tipos de productos (P1, P2, P3) con precios establecidos. Además, ofrece regalos según la cantidad de unidades adquiridas: un lapicero, un cuaderno o una agenda. El programa debe calcular el importe a pagar y determinar el regalo correspondiente, dado el tipo de producto y la cantidad de unidades adquiridas.

Requerimientos Funcionales

1. Entrada de Datos:
 - Tipo de Producto: P1, P2 o P3.
 - Cantidad de Unidades Adquiridas: Entero ≥ 1 .
2. Procesamiento:
 - Determinar el Precio Unitario: Según el tipo de producto.
 - Calcular el Importe a Pagar: Precio unitario \times cantidad.
 - Determinar el Regalo:
 - Menos de 26 unidades: Un lapicero.
 - Entre 26 y 50 unidades: Un cuaderno.
 - Más de 50 unidades: Una agenda.
3. Salida de Datos:
 - Importe a Pagar.
 - Regalo Correspondiente.
4. Validaciones:
 - Verificar que el tipo de producto ingresado sea válido (P1, P2, P3).
 - Asegurar que la cantidad de unidades sea un número entero positivo.

Diagrama de Clases

plaintext

Copiar código

```
+-----+
|  Producto  |
+-----+
| - tipo: String |
| - cantidad: Int |
+-----+
| + obtenerPrecioUnitario() |
| + calcularImporte()      |
| + determinarRegalo()     |
+-----+
```

```
+-----+
|  Regalo    |
+-----+
| - tipoRegalo: String |
+-----+
| + asignarRegalo()    |
+-----+
```

Descripción de Clases:

- Producto:
 - Atributos:
 - tipo: Tipo de producto (P1, P2, P3).
 - cantidad: Número de unidades adquiridas.
 - Métodos:
 - obtenerPrecioUnitario(): Retorna el precio según el tipo.
 - calcularImporte(): Calcula el importe a pagar.
 - determinarRegalo(): Determina el regalo basado en la cantidad.
- Regalo:
 - Atributos:
 - tipoRegalo: Tipo de regalo asignado (lapicero, cuaderno, agenda).

- Métodos:
 - `asignarRegalo()`: Asigna el tipo de regalo según la cantidad de unidades.