



Reto 5 – Call Center

Objetivo:

El objetivo de este reto es que el estudiante reconozca y aplique los elementos básicos del paradigma de la programación orientada a objetos en un escenario abstraído de la cotidianidad. Integrando la lógica de programación y la aplicación de reglas del negocio a sus soluciones.

Contexto:

El **CallCenter** ha decidido incluir una nueva funcionalidad a su sistema de liquidaciones que le permita obtener el costo que le representan sus empleados. Anteriormente se construyeron funcionalidades que permitían obtener el valor de las liquidaciones de nómina, prestaciones sociales, seguridad social y parafiscales de la colección de empleados con la que cuenta el **CallCenter**. Sin embargo, actualmente se requiere de una funcionalidad capaz de llevar a cabo el cálculo de los pagos totales que se le deben otorgar a todos sus empleados, y, además, los cálculos pertinentes a los conceptos de salud, pensión, riesgos labores, caja de compensación, ICBF y SENA, que debe cancelar por ley cuando vincula laboralmente a un **Empleado**. De esta manera, le logrará obtener el costo total de la nómina de empleados con la que cuenta el **CallCenter**.

Reto:

A partir de sus conocimientos en programación orientada a objetos y lógica de programación, desarrolle el método **totalCostoEmpleadosParaLaEmpresa** en la clase **CallCenter**. En este método se dará solución a las necesidades descritas en el contexto del reto. A continuación, se provee un enlace de apoyo que le ayudará a realizar los cálculos que concierne al método **totalCostoEmpleadosParaLaEmpresa**.

Enlace: <https://www.gerencie.com/liquidacion-de-la-nomina.html>

Tenga presente que, para el correcto desarrollo y funcionamiento de la implementación del método **totalCostoEmpleadosParaLaEmpresa** Se deberá contar con las clases **CallCenter** y **Empleado** que se trabajaron en los retos anteriores.



La firma del método **totalCostoEmpleadosParaLaEmpresa** en la clase **CallCenter**, deberá de verse de la siguiente manera:

```
public class CallCenter {  
    ArrayList<Empleado> empleados = new ArrayList<>();  
  
    public static ArrayList<Double> totalCostoEmpleadosParaLaEmpresa(ArrayList<Empleado> empleados){  
        //-- Código Solución --/  
    }  
}
```

Casos de prueba:

Para verificar el correcto funcionamiento de su programa se sugiere que tenga en consideración el siguiente caso de prueba

# CASO DE PRUEBA	DATOS DE ENTRADA	SALIDA ESPERADA		
<i>totalCostoEmpleadosParaLaEmpresa</i>	Vector Vacío	Valor: Vector Vacío		
<i>totalCostoEmpleadosParaLaEmpresa</i>		Valor:		
	Empleado 1	Empleado 2	valor1	Valor2
	Nombre: Carmen	Nombre: Laura	2.338.749,9	2.946.672,8
	Horas Extra: \$40.000	Horas Extra: \$50.000		
	Auxilio: Si	Auxilio: No		
	Salario: \$1'500.000	Salario: \$2'000.000		



Entrega:

1. Suba a la plataforma los archivos **CallCenter.java** y **Empleado.java**, estos nombres deben de respetarse, dado que, si no se nombran de dicha manera no se tendrán en cuenta para la clasificación del reto.
2. **Importante:** El método debe de llamarse **exactamente igual** a como se muestra en la firma de la imagen presentada anteriormente.