



Reto 3 – Empresa Textil

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.

Contexto:

La empresa textil ha ampliado las funcionalidades su sistema de liquidación de nómina y ahora quiere incluir la liquidación de seguridad social y liquidación de prestaciones sociales. Con estas nuevas funcionalidades la **Empresa** se representaría de la siguiente manera:

Empresa
- empleados: ArrayList<Empleado>
+ liquidarNominaEmp(empleados: ArrayList<Empleado>): ArrayList<Double> + liquidarParafiscalesEmp(empleados: ArrayList<Empleado>): ArrayList<Double> + liquidarSegSocialEmp(empleados: ArrayList<Empleado>): ArrayList<Double> + liquidarPrestacionesEmp(empleados: ArrayList<Empleado>): ArrayList<Double> + getters + setters

Anteriormente la empresa contaba con la liquidación de nómina, en donde se toman los conceptos de total devengado (horas extra, auxilio de transporte, salario) y se le deducen la seguridad social que le corresponde al empleado, al igual que llevaba a cabo la liquidación de los parafiscales. Ahora mismo, además de las funcionalidades antes mencionadas, también estará encargada de efectuar la liquidación de la seguridad social y prestaciones sociales de los empleados de su nómina.

Para continuar con la construcción del sistema de liquidaciones, la empresa deberá de deducir del total devengado de un empleado los porcentajes de 8,5% de salud, 12% de pensión y 0,522% de riesgos laborales, para obtener el valor a liquidar por concepto de seguridad social. Adicionalmente, deberá de otorgar a su empleado una bonificación por prestaciones sociales, que corresponde al 8,33% en el caso de la prima de servicios y cesantías, que se encuentran aplicadas sobre el total devengado, 12% de los intereses sobre las cesantías y, por último, un 4,16% de las vacaciones, las cuales se obtiene de aplicar dicho porcentaje al salario del empleado.

Nota: En el caso de la seguridad social, al igual como se realizó previamente en el *Reto2* para los parafiscales, se debe de excluir el concepto de auxilio de transporte del total devengado de un empleado.



Reto:

Adicione los métodos de **liquidarSegSocialEmp** y **liquidarPrestacionesEmp** a la clase **Empresa** con base a la información suministrada en el contexto del reto actual. Tenga en mente que estos métodos tendrán un comportamiento similar a los métodos desarrollados en los retos pasados, por lo que, recibirán una lista de objetos de tipo **Empleado** y para cada uno de ellos retornarán el valor correspondiente a liquidar ya sea de un concepto u otro.

Una vez que incorporé los nuevos métodos a la clase **Empresa** está deberá de verse de la siguiente manera (con respecto a sus funcionalidades):

```
public class Empresa {  
    private ArrayList<Empleado> empleados = new ArrayList<>();  
    public static ArrayList<Double> liquidarNominaEmp(ArrayList<Empleado> empleados){  
    }  
    public static ArrayList<Double> liquidarParafiscalesEmp(ArrayList<Empleado> empleados){  
    }  
    public static ArrayList<Double> liquidarSegSocialEmp(ArrayList<Empleado> empleados){  
    }  
    public static ArrayList<Double> liquidarPrestacionesEmp(ArrayList<Empleado> empleados){  
    }  
}
```



Casos de prueba:

Para verificar el funcionamiento de su programa se sugiere que tenga en consideración los siguientes casos de prueba

# CASO DE PRUEBA	DATOS DE ENTRADA	SALIDA ESPERADA														
liquidarSegSocialEmp	Vector Vacío	Valor: Vector Vacío														
liquidarSegSocialEmp	<table><tr><th>Empleado 1</th><th>Empleado 2</th></tr><tr><td>Nombre: Ada</td><td>Nombre: Laura</td></tr><tr><td>Horas Extra: 5</td><td>Horas Extra: 5</td></tr><tr><td>Auxilio: Si</td><td>Auxilio: Si</td></tr><tr><td>Salario: \$1'500.000</td><td>Salario: \$1'085.000</td></tr></table>	Empleado 1	Empleado 2	Nombre: Ada	Nombre: Laura	Horas Extra: 5	Horas Extra: 5	Auxilio: Si	Auxilio: Si	Salario: \$1'500.000	Salario: \$1'085.000	Valor: <table><tr><th>Valor 1</th><th>Valor 2</th></tr><tr><td>321.899,38</td><td>232.840,55</td></tr></table>	Valor 1	Valor 2	321.899,38	232.840,55
Empleado 1	Empleado 2															
Nombre: Ada	Nombre: Laura															
Horas Extra: 5	Horas Extra: 5															
Auxilio: Si	Auxilio: Si															
Salario: \$1'500.000	Salario: \$1'085.000															
Valor 1	Valor 2															
321.899,38	232.840,55															
liquidarPrestacionesEmp	Vector Vacío	Valor: Vector Vacío														
liquidarPrestacionesEmp	<table><tr><th>Empleado 1</th><th>Empleado 2</th></tr><tr><td>Nombre: Ada</td><td>Nombre: Laura</td></tr><tr><td>Horas Extra: 5</td><td>Horas Extra: 5</td></tr><tr><td>Auxilio: Si</td><td>Auxilio: Si</td></tr><tr><td>Salario: \$1'500.000</td><td>Salario: \$1'085.000</td></tr></table>	Empleado 1	Empleado 2	Nombre: Ada	Nombre: Laura	Horas Extra: 5	Horas Extra: 5	Auxilio: Si	Auxilio: Si	Salario: \$1'500.000	Salario: \$1'085.000	Valor: <table><tr><th>Valor 1</th><th>Valor 2</th></tr><tr><td>351.611,98</td><td>259.533,82</td></tr></table>	Valor 1	Valor 2	351.611,98	259.533,82
Empleado 1	Empleado 2															
Nombre: Ada	Nombre: Laura															
Horas Extra: 5	Horas Extra: 5															
Auxilio: Si	Auxilio: Si															
Salario: \$1'500.000	Salario: \$1'085.000															
Valor 1	Valor 2															
351.611,98	259.533,82															

Entrega:

1. Suba a la plataforma los archivos **Empresa.java** y **Empleado.java**, estos nombres deben de respetarse, dado que, si no se nombran de dicha manera no se tendrá en cuenta para la calificación del reto.
2. **Importante:** Los métodos deben de llamarse **exactamente igual** a como se muestra en el ejemplo de la estructura del código.