Reto 3 – Banco

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:

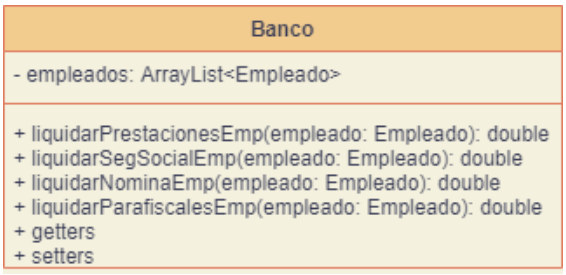
Previamente el sistema de liquidaciones del **Banco**, se efectuaban los cálculos de la liquidación de prestaciones sociales y seguridad social de un empleado. Actualmente, se requiere añadir 2 nuevas funcionalidades que le proporcionen al sistema de liquidación una mayor operabilidad. Estas funcionalidades se describen a continuación:

- **Liquidación de nómina:** Para llevar a cabo su estimación, se deben de sumar los conceptos de salario base, comisiones, horas extras y auxilio de transporte, en caso de que el empleado los presente, y a ese valor deducirle la salud y pensión que le corresponden al empleado.

  Nota: Los pagos que le corresponde al empleado y deben ser descontados de su nómina, son 4% por salud y 4% por pensión de su total devengado. Sin embargo, para efectos de la seguridad social no se tiene en cuenta el auxilio de transporte, y se debe excluir del cálculo.

- **Liquidación de parafiscales:** La liquidación de los parafiscales se obtiene a partir de sumar las deducciones que van dirigidas a la caja de compensación familiar, ICBF y SENA**.** Estas corresponden a los porcentajes de 4%, 3% y 2% respectivamente, y similar a la seguridad social se descuentan del total devengado de un empleado, pero excluyendo el auxilio de transporte que no es un factor salarial.

Incorporando estas 2 nuevas funcionalidades la representación de la clase **Banco** en el sistema se varía de la siguiente manera:



Reto:

Se requiere que implemente los métodos **liquidarNominaEmp** y **liquidarParafiscalesEmp** en la clase **Banco** para continuar ampliando la lógica del sistema de liquidaciones. Semejante a los métodos desarrollados con anterioridad en el *Reto2*, se recibirá como parámetro un objeto de la clase **Empleado** y se retornará el valor correspondiente a ese empleado. Para efectuar el cálculo de las liquidaciones, básese en las descripciones brindadas en el contexto del reto en cuestión.

Con la adición de los nuevos métodos, la clase **Banco** vista en forma de código (implementación), presentaría los comportamientos mostrados a continuación:

```
public class Banco {

  ArrayList<Empleado> empleados = new ArrayList<>();

  public static double liquidarPrestacionesEmp(Empleado empleado){

  }

  public static double liquidarSegSocialEmp(Empleado empleado){

  }

  public static double liquidarNominaEmp(Empleado empleado){

  }

  public static double liquidarParafiscalesEmp(Empleado empleado){

  }
}
```

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
liquidarNominaEmp	**Empleado** Nombre: Manuel Salario: $2’500.000	***Valor:*** 2.300.000,00
liquidarParafiscalesEmp	**Empleado** Nombre: Manuel Salario: $2’500.000	***Valor:*** 225.000,00

Entrega:

1. Suba a la plataforma los archivos **Banco.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.