

### **Simulacro Primer Parcial: Venta Electrodomésticos**

Una financiera de una conocida firma que vende electrodomésticos, desea guardar y gestionar información de los prestamos que otorga. Para ello la financiera guarda la colección de prestamos que fueron otorgados y cada una de sus cuotas.

Implementar la clase **Financiera** , la clase **Préstamo**, la clase **Cuota** y la clase **Persona** .

En la clase **Persona**:

1. Se registra la siguiente información: nombre, apellido, dni, direccion, mail, telefono y neto.
2. Método constructor que recibe como parámetros los valores iniciales para los atributos definidos en la clase.
3. Los métodos de acceso de cada uno de los atributos de la clase.
4. Redefinir el método **\_toString** para que retorne la información de los atributos de la clase.

En la clase **Cuota**:

1. Se registra la siguiente información: número ,monto\_cuota , monto\_interes y cancelada (atributo que va a contener un valor true, si la cuota esta paga y false en caso contrario)
2. Método constructor que recibe como parámetros los valores iniciales para los atributos: número, monto\_cuota y monto\_interes definidos en la clase. Por defecto todas las cuotas deben ser generadas como canceladas = false.
3. Los métodos de acceso de cada uno de los atributos de la clase.
4. Redefinir el método **\_toString** para que retorne la información de los atributos de la clase.
5. Implementar el método **darMontoFinalCuota()** que retorna el importe total de la cuota mas los intereses que deben ser aplicados.

En la clase **Prestamo**:

1. Se registra la siguiente información: identificación, código del electrodoméstico, fecha otorgamiento, monto, cantidad\_de\_cuotas, tasa de interés, la colección de cuotas y la referencia a la persona que solicito el préstamo.
2. Método constructor que recibe como parámetros los siguientes valores: identificación, monto, cantidad de cuotas, tasa de interés y la referencia a la persona que solicito el préstamo. El constructor debe asignar los valores recibidos a las variables instancias que corresponda.
3. Los métodos de acceso de cada uno de los atributos de la clase.
4. Redefinir el método **\_toString** para que retorne la información de los atributos de la clase.
5. Implementar el método privado **calcularInteresPrestamo(numCuota)** que recibe por parámetro el número de la cuota y calcula el importe del interés sobre el saldo deudor.

Por ejemplo si el préstamo tiene 5 cuotas, el monto total = 50000 y el interés 0.1% entonces el monto del interés sobre saldo deudor que debe calcularse para cada una de las cuotas deben ser los siguientes:

- interés cuota 1:  $50\ 000 * 0.01 = 500$
- interés cuota 2:  $(50\ 000 - (50\ 000 / 5)) * 0.01 = 400$
- interés cuota 3:  $(50\ 000 - ((50\ 000 / 5) * 2)) * 0.01 = 300$
- interés cuota 4:  $(50\ 000 - ((50\ 000 / 5) * 3)) * 0.01 = 200$
- interés cuota 5:  $(50\ 000 - ((50\ 000 / 5) * 4)) * 0.01 = 100$

interes cuota **numCuota** =  $(\text{monto} - ((\text{monto} / \text{cantidad\_de\_cuotas}) * \text{numCuota} - 1)) * \text{taza\_de\_interés} / 0.01$

6. Implementar el método **otorgarPrestamo** que setea la variable instancia **fecha otorgamiento**, con la fecha actual (puede utilizar el valor retornado por la función de PHP **getdate()**) y genera cada una de las cuotas dependiendo el valor almacenado en la variable instancia “cantidad\_de\_cuotas” y **monto**. El importe total de la cuota debe ser calculado de la siguiente manera:  $(\text{monto} / \text{cantidad\_de\_cuotas})$  y el monto correspondiente al interés debe ser el valor retornado por el método **calcularInteresPrestamo(numeroCuota)** implementado en el inciso anterior.

7. Implementar el método **darSiguienteCuotaPagar** método que retorna la referencia a la siguiente cuota que debe ser abonada de un préstamo, si el préstamo tiene todas sus cuotas canceladas retorna null.

En la clase **Financiera**:

1. Se registra la siguiente información: denominación, dirección y la colección de prestamos otorgados.
2. Método constructor que recibe como parámetros los valores iniciales para los atributos de la clase denominación, dirección.
3. Los métodos de acceso para cada una de las variables instancias de la clase.
4. Redefinir el método **\_toString** para que retorne la información de los atributos de la clase.
5. Implementar el método **incorporarPrestamo** que recibe por parámetro un nuevo préstamo.
6. Implementar el método **otorgarPrestamoSiCalifica**, método que recorre la lista de prestamos que no han sido generadas sus cuotas. Por cada préstamo, se corrobora que su monto dividido la cantidad\_de\_cuotas no supere el 40 % del neto del solicitante, si la verificación es satisfactoria se invoca al método **otorgarPrestamo**.
7. Implementar el método **informarCuotaPagar(idPrestamo)** que recibe por parámetro la identificación del préstamo, se busca el préstamo en la colección de prestamos y si es encontrado se obtiene la siguiente cuota a pagar. El método debe retornar la referencia a la cuota. Utilizar para su implementación el método **darSiguienteCuotaPagar**

Implementar un script **TestFinanciera** en la cual:

1. Se crea un objeto Financiera con la siguiente información: denominación= **Money**, dirección = **"Av. Arg 1234 "**
2. Se crea 3 objetos Prestamos con la información visualizada en la tabla:

identificación	monto	cantidad_de_cuotas	taza_de_interés	Persona (nombre, apellido, direccion, mail , telefono, neto )
1	50000	5	0.1	Pepe , Florez, Bs As 12 , <a href="mailto:dir@mail.com">dir@mail.com</a> , 299 444567, 40000
2	10000	4	0.1	Luis , Suarez, Bs As 123 , <a href="mailto:dir@mail.com">dir@mail.com</a> , 299 4455, 4000
3	10000	2	0.1	Luis , Suarez, Bs As 123 , <a href="mailto:dir@mail.com">dir@mail.com</a> , 299 4455, 4000

3. Invocar al método **incorporarPrestamo** de la Clase Financiera con cada uno de los prestamos creados en el inciso anterior.
4. Realizar un echo del objeto Financiera creado en 1).
5. Invocar al método **otorgarPrestamoSiCalifica** de la Clase Financiera.
6. Realizar un echo del objeto Financiera creado en 1).
7. Invocar al método **informarCuotaPagar(2)** de la Clase Financiera y almacenar el valor en una variable \$objCuota.
8. Realizar un echo de la variable obtenida en el inciso anterior.
9. Invocar al método **darMontoFinalCuota** con el objeto obtenido en el inciso 7 y visualizar el resultado obtenido
10. Invocar al método **setCancelada(true)** con el objeto obtenido en el inciso 7.
11. Invocar al método **informarCuotaPagar(2)** de la Clase Financiera y almacenar el valor en una variable \$objCuota.
12. Realizar un echo de la variable obtenida en el inciso anterior.