

TITULO

PROGRAMACION ORIENTADA A OBJETOS

Semana 04

CLASES, OBJETOS Y ATRIBUTOS

EJERCICIO

1. Desarrolle una aplicación móvil que permita almacenar y mostrar el nombre, apellidos, código y el promedio de un alumno , además mostrar si el alumno a aprobó o desaprobado, aplicar POO utilizando dos clases.



EJERCICIO

- 2.- Desarrolle una aplicación móvil que permita crear la clase Profesor, que tenga como atributos código, nombre, apellido, curso que enseña, numero de horas dictadas, tarifa por hora y sueldo, además tendrá los siguientes métodos:
- **asignarDatos**, donde recibe como parámetro el nombre, apellido , curso, num. Horas y tarifa por hora.
 - **generarCodigo**, el código sea un autogenerado de 6 dígitos.
 - **retornarDatos** que retorna todos los datos del profesor.
 - **calcularSueldo**, multiplicar la tarifa por numero de horas dictadas.

Diseñe una pantalla para ingresar y mostrar los datos del profesor , además cree un objeto que instancie a la clase Profesor.

EJERCICIO

3.- Desarrolle una aplicación móvil que almacene y muestre los datos de 3 trabajadores de una empresa textil. Considere que los datos son: código , apellidos, nombres y el sueldo que se asigna de acuerdo a categoría como se muestra en la tabla.

Categoría	Sueldo en S/.
A	1000
B	1500
C	2500
D	3500

Crea la clase **Trabajador** con todos los atributos y los siguientes métodos:

- **asignarDatos.**
- **retornarDatos**
- **calcularSueldo**

Cree los objetos que crea conveniente para almacenar los datos de los Trabajadores.

EJERCICIO

4.- Desarrolle una aplicación móvil que permita crear una clase **Ciente** que consiste en varios miembros de datos, como numero de cuenta, apellidos, nombres y saldo, además tendrá los siguientes métodos:

- `calcularSaldo()` que retorne el saldo, considere que el saldo que tiene la cuenta antes de realizar el primer movimiento es autogenerado en el rango de [1000 – 50000] soles.
- `retirarMonto()`, el monto mínimo que un cliente puede retirar es de S/ 20 soles y el mayor monto es de S/500 soles, además el monto a retirar tiene que ser múltiplo de 20, 50 o 100.
- `depositarMonto()`, si el cliente hace un deposito como mínimo tiene que depositar S/. 100 soles.

Diseñe una pantalla `lytcliente` que permita crear los objetos, ingresar y mostrar los datos.

5.- Desarrolle una aplicación móvil que permita crear una clase **CajeroAutomatico** que consiste en varios miembros de datos, como numero de cuenta, apellidos, nombres, edad y saldo, además tendrá los siguientes métodos:

- `calcularSaldo()` que retorne el monto del saldo, considere que el saldo que tiene la cuenta antes de realizar el primer movimiento es autogenerado en el rango de acuerdo a la edad del cliente :

Edad	Rango
[16 – 18]	[1000 – 10000]
[19 – 30]	[10001-25000]
[31- 100>	[25001-50000]

- `retirarMonto()`, el monto mínimo que un cliente puede retirar es de S/ 20 soles, el monto máximo es de S/500 soles si es menor de edad, caso contrario puede retirar hasta S/ 1,500 soles, considere que el cajero solo puede dar billetes de 10,20, 50, 100 o 200 soles.
- `depositarMonto()`, si el cliente hace un deposito como mínimo tiene que depositar S/. 100 soles, considere que el cajero solo recibe billetes de 50, 100 o 200 soles.

Diseñe la clase **FrmCliente** que permita crear los objetos, ingresar y mostrar los datos.