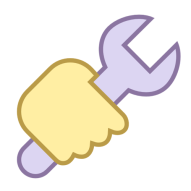
**CFP 8**

**Módulo 2:**

**PROGRAMACIÓN ORIENTADA A OBJETOS**

**Unidad 1:**

**INTRODUCCIÓN AL PARADIGMA DE OBJETOS**

****

## ACTIVIDADES PRÁCTICAS

### Clases y Objetos, Atributos y Comportamiento, Estado de un objeto, Identidad, Constructores

**1)**

¿Cuál es la instrucción Java que debemos usar para crear la clase Alumno?

Public class Alumno{

//atributos

//constructores

//metodos

}

**2)**

Dado el siguiente código, indicar cuál es el método constructor y cómo se identifica.

**public** **class** PuebloDelOeste

{

**int** establos;

**int** cantinas;

**int** comisarios;

**int** alborotadores;

String locacion;

**int** tiempo;

//este es el método constructor y se identifica porque se llama igual que la clase

**public** PuebloDelOeste()

{

establos = 3;

locacion = "Oeste de los Estados Unidos";

tiempo = 1850;

}

}

**3)**

¿Por qué decimos que esta clase es ejecutable?

//porque esta declarada dentro del método main, sabiendo que el método main es el método que nos permite ejecutar la aplicación o todo lo allí delclarado.

**public** **class** Balacera

{

**public** **static** **void** main (String args[])

{

PuebloDelOeste sweatyPost = **new** PuebloDelOeste();

sweatyPost.cantinas = 2;

sweatyPost.comisarios = 1;

sweatyPost.alborotadores = 5;

}

}

**4)**

¿Qué diferencia existe entre el método **Humanos** y el método **comoTeLlamas**?

**//humanos NO es un método, es una clase que contiene al método comoTeLlamas() que a su ves devuelve el nombre de el humano**

**public** **class** Humanos {

**int** piernas;

**int** brazos;

**int** ojos;

**int** nariz;

**int** boca;

String nombre;

String sexo;

String nombreCaballo;

String preferenciaWhiskey;

**public** Humanos() {

piernas = 2;

brazos = 2;

ojos = 2;

nariz = 1;

boca = 1;

}

**public** String comoTeLlamas() {

**return** nombre;

}

}

**5)**

Dado el siguiente código, escribir el método constructor.

**public** **class** Persona

{

**private** String nombre;

**private** **double** altura;

**private** **int** edad;

**private** String genero;

Public Persona(){};

public Persona (String nombre, **double** altura, **int** edad, String genero){

this.nombre = nombre;

this.altura = altura;

this.edad = edad;

this.genero = genero;

}

**public** **int** getEdad() {

**return** edad;

}

}

**6)**

Crear la clase Gato en java, teniendo en cuenta que sus atributos son: nombre, raza, color y sus métodos: maullar(), caminar(), saltar(), jugar().

public class Gato {

private String nombre;

private String raza;

private String color;

public Gato() {

}

public Gato(String nombre, String raza, String color) {

this.nombre = nombre;

this.raza = raza;

this.color = color;

}

public String getColor() {

return color;

}

public void setColor(String color) {

this.color = color;

}

public String getRaza() {

return raza;

}

public void setRaza(String raza) {

this.raza = raza;

}

public String getNombre() {

return nombre;

}

public void setNombre(String nombre) {

this.nombre = nombre;

}

//metodos

public void maullar() {

System.out.println("MIAU");

}

public void caminar() {

System.out.println("vamos a caminar");

}

public void saltar() {

System.out.println("saltemos un poco");

}

public void jugar() {

System.out.println("arroja algo, dale, quiero jugar");

}

@Override

public String toString() {

return "nombre=" + nombre + ", raza=" + raza + ", color=" + color ;

}

}

**7)**

Crear una clase llamada Motocicleta con los siguientes atributos:

* Matrícula: String
* Color: String
* Velocidad: entero
* enMarcha: booleana

public class Motocicleta{

public String matricula;

public String color;

public int velocidad;

public boolean enMarcha;

public void arrancar(){

}

public void acelerar(){

this.velocidad \*= 2;

}

public void frenar(){

this.velocidad -= 10;

}

public void girar(){

sout(“estoy girando”);

}

}

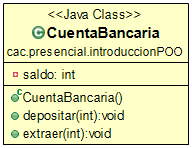
**8)**

En base a la clase Motocicleta creada en el ejercicio anterior, crearle los siguientes métodos:

* arrancar
* acelerar
* frenar
* girar

**9)**

Crear la clase CuentaBancaria, con el atributo y los métodos que figuran a continuación.



public class CuentaBancaria {

private int saldo;

public CuentaBancaria(int saldo) {

this.saldo = saldo;

}

public int getSaldo() {

return saldo;

}

public void setSaldo(int saldo) {

this.saldo = saldo;

}

public depositar(int saldo) {

this. saldo += saldo;

}

public extraer(int monto) {

this.saldo += monto;

}

}

**10)**

La empresa informática “IPM Tech” necesita llevar un registro de todos sus empleados que se encuentran en la oficina central, para eso necesita implementar la clase Empleado que debe tener los siguientes atributos:

* nombre: cadena (nombre y apellido)
* cedula: cadena
* edad : entero (entre 18 y 45 años)
* casado: booleano
* salario: numérico doble

Y los siguientes métodos:

* Constructor con y sin parámetros de entrada.
* Método que permita visualizar la clasificación según la edad del empleado de acuerdo al siguiente algoritmo:
  + Si edad es menor a 21, Junior
  + Si edad se encuentra entre 21 y 34, Intermedio
  + Si edad es mayor o igual a 35, Senior.
* Imprimir los datos del empleado por pantalla (se puede utilizar salto de línea \n para separar los atributos.
* Un método que permita aumentar el salario en un porcentaje que podrá ser pasado como parámetro al método.
* **Ayuda:** crear un método **main** para poder crear el objeto “emp1” e imprimir con él sus valores con **System.out**.

public class Persona {

private String nombre;

private String apellido;

public Persona(String nombre, String apellido) {

this.nombre = nombre;

this.apellido = apellido;

}

public String getApellido() {

return apellido;

}

public void setApellido(String apellido) {

this.apellido = apellido;

}

public String getNombre() {

return nombre;

}

public void setNombre(String nombre) {

this.nombre = nombre;

}

@Override

public String toString() {

return "Persona{" + "nombre=" + nombre + ", apellido=" + apellido + '}';

}

}

public class Empleado {

private Persona nombre;

private int dni;

private int edad;

public int getEdad() {

return edad;

}

public void setEdad(int edad) {

this.edad = edad;

}

private boolean casado;

private double salario;

public Empleado() {

}

public Empleado(Persona nombre, int dni, boolean casado, double salario, int edad) {

this.nombre = nombre;

this.dni = dni;

this.casado = casado;

this.salario = salario;

this.edad = edad;

}

public double getSalario() {

return salario;

}

public void setSalario(double salario) {

this.salario = salario;

}

public boolean isCasado() {

return casado;

}

public void setCasado(boolean casado) {

this.casado = casado;

}

public int getDni() {

return dni;

}

public void setDni(int dni) {

this.dni = dni;

}

public Persona getNombre() {

return nombre;

}

public void setNombre(Persona nombre) {

this.nombre = nombre;

}

public String clasificarSegunEdad() {

if (this.edad<21) {

return "menor";

}else{

if ((this.edad >= 21) && (this.edad < 34)) {

return "intermedio";

}else{

return "mayor";

}

}

}

public double aumentarSalario(float procentaje) {

return this.salario\*procentaje;

}

@Override

public String toString() {

return "nombre=" + nombre + ", dni=" + dni + ", edad=" + edad + ", casado=" + casado + ", salario=" + salario;

}

}

public class Test {

public static void main(String[] args) {

Empleado emp1 = new Empleado(new Persona("manuel", "soto"), 12345, true, 10000, 35);

System.out.println(emp1);

}

}