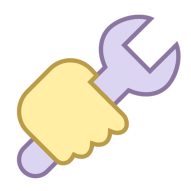
**CFP 8**

**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?

**Respuesta: public class Alumno { }**

**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;

**public** PuebloDelOeste()

{

establos = 3;

locacion = "Oeste de los Estados Unidos";

tiempo = 1850;

}

}

**Respuesta:**

**El método constructor es :**

**public** PuebloDelOeste()

{

establos = 3;

locacion = "Oeste de los Estados Unidos";

tiempo = 1850;

}

**Se puede identificar como el método constructor por que se utiliza para inicializar los atributos de la clase cuando se crea una instancia de la misma.**

**Tiene el mismo nombre de la clase y no cuenta con ningun valor de retorno, ya que su función principal es inicializar el objeto y no devolver ningún valor.**

**3)**

¿Por qué decimos que esta clase es ejecutable?

**public** **class** Balacera

{

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

{

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

sweatyPost.cantinas = 2;

sweatyPost.comisarios = 1;

sweatyPost.alborotadores = 5;

}

}

**Respuesta: Porque tiene el método main.**

**4)**

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

**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;

}

}

**Respuesta: La diferencia entre el método Humanos y el método comoTeLlamas, es que el primero es un método constructor (Humanos) y el segundo (comoTeLlamas) es un método de instancia común .**

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

**return** edad;

}

}

**Respuesta: public Persona() { }**

**6)**

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

**Respuesta:**

public class Gato {

public String nombre;

public String raza;

public String color;

public Gato() { }

public void maullar () { }

public void caminar () { }

public void saltar() { }

public void jugar () { }

}

**7)**

Crear una clase llamada Motocicleta con los siguientes atributos:

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

**Respuesta:**

public class Motocicleta {

public String Matricula;

public String Color;

public int Velocidad;

public boolean enMarcha ;

}

**8)**

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

* arrancar
* acelerar
* frenar
* girar

**Respuesta:**

public class Motocicleta {

public String Matricula;

public String Color;

public int Velocidad;

public boolean enMarcha ;

public void arrancar() { }

public void acelerar() { }

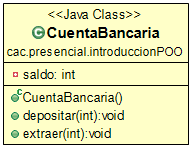
public void frenar() { }

public void girar () { }

}

**9)**

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



**Respuesta:**

public class CuentaBancaria {

private int saldo;

public CuentaBancaria(int saldoInicial) {

saldo = saldoInicial;

}

public void depositar(int monto) {

if (monto > 0) {

saldo += monto;

}

else {

System.out.println("El monto a depositar debe ser mayor que cero.");

}

}

public void extraer(int monto) {

if (monto > 0 && monto <= saldo)

{

saldo -= monto;

}

else {

System.out.println("Monto insuficiente o inválido para retirar.");

}

}

public int getSaldo() {

return saldo;

}

}

**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. (vacio / cargado)
* 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**.

**Respuesta:**

public class Empleado {

private String nombre;

private String cedula;

private int edad;

private boolean casado;

public double salario;

public Empleado() { }

public Empleado(String nombre, String cedula, int edad, boolean casado, double salario) {

this.nombre = nombre;

this.cedula = cedula;

this.setEdad(edad);

this.casado = casado;

this.salario = salario;

}

public String clasificarEmpleado() {

if (edad < 21)

{

return "Junior";

}

else if (edad >= 21 && edad <= 34)}

{

return "Intermedio";

}

else {

return "Senior";

}

}

public void imprimirDatos() {

System.out.println("Nombre: " + nombre);

System.out.println("Cédula: " + cedula);

System.out.println("Edad: " + edad);

System.out.println("Casado: " + casado);

System.out.println("Salario: " + salario);

}

public void aumentarSalario(double porcentaje) {

salario = salario \* (1 + porcentaje / 100);

}

public int getEdad() {

return edad;

}

public void setEdad(int edad) {

if (edad >= 18 && edad <= 45) {

this.edad = edad;

} else {

System.out.println("La edad debe estar entre 18 y 45 años.");

}

}

}