

CENTRO DE TECNOLOGÍA Y EDUCACIÓN  
PERMANENTE



DIPLOMADO EN ARQUITECTURA Y DISEÑO DE SOLUCIONES WEB Y MÓVIL

**Asignación:**

Reporte Practica 1

**Entregado a:**

Ing. Carlos Camacho

**Entregado por:**

Dariana Tavera

Santiago, Sábado 19 de marzo 2016

# Introducción

Java es un lenguaje de programación orientado a objeto, fuertemente tipado. Este tiene como filosofía 'escribe una vez y corre en todas partes', esto conjunto su flexibilidad, escalabilidad y fuente abierta constituyen la posible razón por la cual se encuentra en el primer lugar en la mayoría de los tops mundiales de lenguajes de desarrollo.

Las clases, los objetos, los tipos y la interrelación entre estos, constituyen la base para desarrollar aplicaciones en Java.

En el siguiente reporte se muestran el desarrollo de ejercicios propuestos para asentar los conocimientos básicos del lenguaje.

## Desarrollo de la práctica:

1. El programa Verificar debería escribir por pantalla las siguientes líneas, pero tiene errores; por lo cual debe corregir los mismos.

En este ejercicio se corrigieron errores de sintaxis, algunos fueron:

- Cadena de caracteres dentro de comillas simples.
- Falta de punto y coma (;) al final de la expresión.
- Forma incorrecta de concatenar los parametros a imprimir.
- Declaracion de variables de tipo primitivo duplicada.

2. Escriba una clase que obtenga 3 cantidades por parámetro y desplegar un mensaje si las cantidades introducidas están en orden numérico, de lo contrario indique del mismo modo como debe de entrar las cantidades para que estén dicho orden.

Se realizó una clase denominada Ejercicio2 y esta contiene el main. Para obtener los valores introducidos es utilizado un objeto Scanner. El algoritmo consiste en tomar los 3 valores introducidos organizarlos de menor a mayor y luego comparar si fue el orden en el que el usuario introdujo los datos.

3. Escribir un programa en java que declare una clase llamada Estudiante y que tenga todas las propiedades y métodos que pueda tener un estudiante. Instanciar esta clase 5 veces en otra clase que contenga el método main y cambie las propiedades a cada estudiante. Recuerde declarar métodos para realizar dicha operación.

Se creó la clase estudiante como se podrá apreciar en el código de la aplicación.

En esta coloque las propiedades que considere necesarias para definir un estudiante. Puse algunos constructores para que existan diferentes métodos para inicializar, algunos métodos que representan acciones realizadas por los estudiante y se incluyeron los métodos de setter y getter.

4. Escribir una clase llamada Edificio que en el constructor se le pueda pasar la cantidad de pisos que tendrás, el nombre y el número de puerta y que existan métodos que permitan cambiar cada uno dichas propiedades e imprimir cada uno en pantalla.

Se creó la clase con los atributos requeridos, todos estos públicos para que pudieran ser accedidos sin problema, se incluyeron los métodos setters y getters y otros métodos que permiten imprimir la información.

5. Escriba una clase Java que representa un círculo; el círculo queda perfectamente definido si se conoce su radio (pasarlo por el constructor como double). Defina además, para esta clase, dos métodos (públicos) que permitan calcular el área del círculo y el perímetro de la circunferencia que delimita el círculo. Para probar la funcionalidad antes definida, escriba un pequeño programa que cree un círculo con un radio dado, y que calcule (y muestre por pantalla) el área y el perímetro de su circunferencia.

Se creó la clase con un solo constructor para pasarle el parámetro del radio. El atributo radio tiene el modificador de acceso private debido a que no es necesario que el usuario lo modifique directamente.

6. Se quiere definir una clase que permita controlar un sintonizador digital de emisoras FM; concretamente, lo que se desea es dotar al controlador de una interfaz que permita subir (up) o bajar (down) la frecuencia (en saltos de 0.5 MHz) y mostrar la frecuencia sintonizada en un momento dado (display). Supondremos que el rango de frecuencias a manejar oscila entre los 80 Mhz y los 108 MHz y que al inicio, el controlador sintoniza a 80MHz. Si durante una operación de subida o bajada se sobrepasa uno de los dos límites, la frecuencia sintonizada debe pasar a ser la del extremo contrario. Realice la clase que contenga dos métodos Up para subir la frecuencia y Down para bajarla, del mismo modo un método que imprima la frecuencia donde se encuentra el radio.

En esta clase el único atributo era la frecuencia, con un solo constructor que es el vacío, en el cual se le asigna a la frecuencia el valor de 80.

Para redondear los datos double utilice un objeto DecimalFormat.

7. Crear una clase "Mi\_Clase" que posea un atributo de clase llamado "valor\_clase" y otro de instancia denominado "valor\_instancia". Debe tener los métodos correspondientes para modificar y obtener el estado de cada uno de los atributos mencionados. Para el atributo "valor\_clase" se debe incrementar por cada objeto creado -inicialmente inicializado en cero. Codificar una aplicación Java y

crear varios objetos pertenecientes a esta clase de manera que se pueda observar el valor de los atributos durante la ejecución de la aplicación.

Cree una clase “Mi\_Clase” en la cual cree los atributos requeridos de la siguiente forma:

```
public static int valor_clase =0;// Este representa un valor  
int valor_instancia; // Este se pasara por referencia
```

Los métodos que utilice para establecer y obtener los valores, fueron getters y setters.

8. Queremos modelar una casa con muchas bombillas, de forma que cada bombilla se puede encender o apagar individualmente. Para ello haremos una clase Bombilla con una variable privada que indique si está encendida o apagada, así como un método que nos diga si una bombilla concreta está encendida. Además queremos poner un interruptor general de la luz, tal que si saltan los fusibles, todas las bombillas quedan apagadas. Cuando el fusible se repara, las bombillas vuelven a estar encendidas o apagadas, según estuvieran antes del percance. Para modelar en Java esta característica usaremos una variable de clase (estática) que nos diga si hay luz en el edificio o no. Cada objeto Bombilla se enciende y se apaga individualmente; pero sólo responde que está encendida si su interruptor particular está activado y además hay luz general.

La clase se implementara con un indicador de estado, que permitirá ver de forma individual si un bombillo está apagado o encendido, el interruptor general se representará con una variable estática debido a que el valor de esta será el mismo para todos los objetos de la clase.

Los métodos para controlar las bombillas de forma individual son encender() y apagar() y el que muestra el estatus general de las bombillas es verEstado() que devuelve un String que indica si esta encendido o apagado.

## Conclusión

En los Ejercicios anteriores se implementaron los conceptos básicos de Java, entre estos están:

- Sintaxis del lenguaje Java.
- Variables y tipado.
- Estructuras de control.
- Modificadores de acceso.

La parte que me resulto mas dificil de implementar fue la de los modificadores de acceso debido a confusiones conceptuales.