

ENTERPRISE JAVA DEVELOPER

JAVA ORIENTADO A OBJETOS

Variables y Métodos de Instancia y de Clase

Eric Gustavo Coronel Castillo
gcoronelc.blogspot.com



Temas



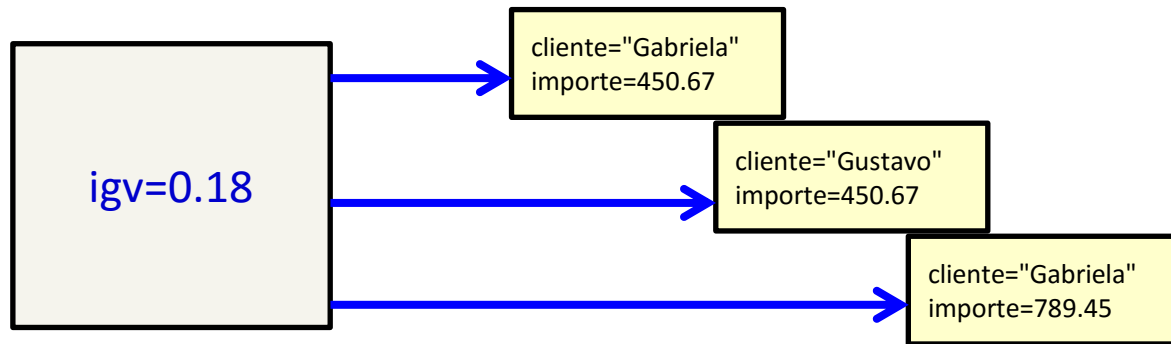
- Objetivo
- Definiciones
- Implementación
- Acceso a las Variables y Métodos
- Inicializador Estático
- Proyecto Ejemplo



OBJETIVO

- Entender la diferencia entre variables y métodos de instancia y de clase.
- Aplicar correctamente variables y métodos de instancia y de clase.

```
public class Venta{  
    private static double igv = 0.18;    // Variable de clase  
    private String cliente;              // Variable de instancia  
    private double importe;              // Variable de instancia  
    ...  
}
```



Variable de la clase **Venta**

Objetos de tipo **Venta** tienen sus propias variables



DEFINICIONES

VARIABLES Y METODOS DE CLASE

- Se trata de variables y métodos que no requieren crear una instancia (objeto) para ser invocados, basta con anteponer el nombre de la clase para poder acceder a ellos.
 - NombreClase.variable
 - NombreClase.método(...)

VARIABLES Y METODOS DE INSTANCIA

- Se trata de variables y métodos que se crean en el objeto y por lo tanto para ser invocados se necesita una instancia de la clase, es decir, un objeto.
 - objeto.variable
 - objeto.método(...)



IMPLEMENTACIÓN

VARIABLES

[visibilidad] [**static**] tipo campo [=valor];

MÉTODOS

[visibilidad] [**static**] tipo nombreMétodo(...) {

 // Implementación

 [return valor;]

}

La presencia de la palabra **static** determina que la declaración es de la clase y no de la instancia.



ACCESO A LAS VARIABLES Y MÉTODOS

DE INSTANCIA

- Desde la misma clase

`this.variable`
`this.método(...)`

- Desde fuera de la clase

`objeto.variable`
`objeto.método(...)`

DE CLASE

- Desde la misma clase

`variable`
`método(...)`

- Desde fuera de la clase

`NombreClase.variable`
`NombreClase.método(...)`



INICIALIZADOR ESTÁTICO

```
public class NombreClase {
```

```
    . . .
```

```
    . . .
```

```
    static {
```

```
        . . .
```

```
        . . .
```

```
    }
```

```
    . . .
```

```
    . . .
```

```
}
```

Se utiliza para inicializar variables de clase, similar a un constructor que se utiliza para inicializar las variables de instancia.



PROYECTO EJEMPLO

El colegio "Ángeles del Cielo" esta solicitando un programa en Java para que los alumnos de primaria verifiquen sus ejercicios de matemáticas referidos a:

- Calculo de factorial
- Calculo del MCD y MCM de dos números
- La serie de Fibonacci
- Número primo

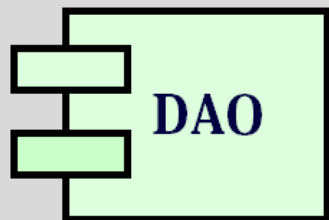
La programación de estos cálculos matemáticos deben estar implementados como métodos de clase en una clase de nombre **MyMath**.



CODIGO FUENTE

EUREKA-CS-ORACLE-JDBC

APLICACIÓN JAVA



JDBC

CONEXIÓN

Esquema

EUREKA

ORACLE XE 11g

Dirección de descarga: <https://goo.gl/TDgc5R>

ENTERPRISE JAVA DEVELOPER

JAVA ORIENTADO A OBJETOS

Gracias

Eric Gustavo Coronel Castillo
gcoronelc.blogspot.com

