

Fundamentos de Programación

Clase 2

Agenda

- Flujo de programa
- Variables
- Lenguaje Java
- Sintaxis
 - Clases
 - Comentarios
 - Tipos de datos
- Clases y objetos
- Practica
- Estructura de un programa en Java
- Instrucciones básicas en Java

Flujo de Programa

- Un **algoritmo** dentro de un programa se compone de **instrucciones**
- El **compilador** ejecuta cada **instrucción** de arriba hacia abajo
- Cada **lenguaje** tiene su propia sintaxis para escribir instrucciones

```
int a = 5;  
int b = 3;  
int c = a + b;  
System.out.println(c);
```

5

a : int

3

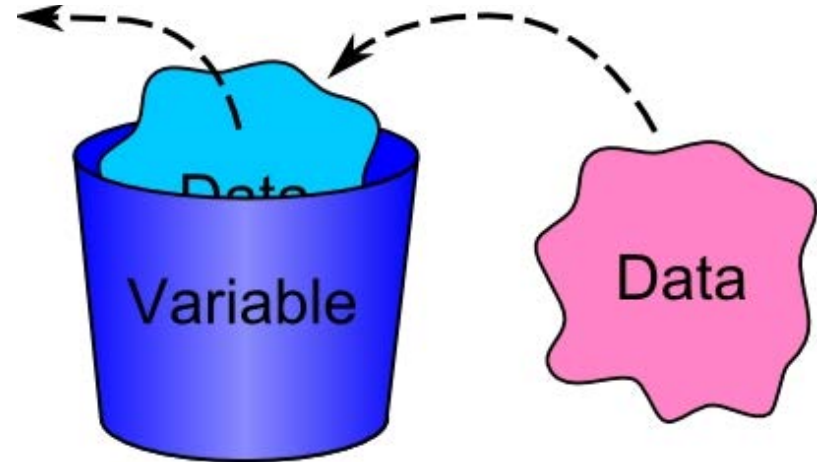
b : int

8

c : int

Variables

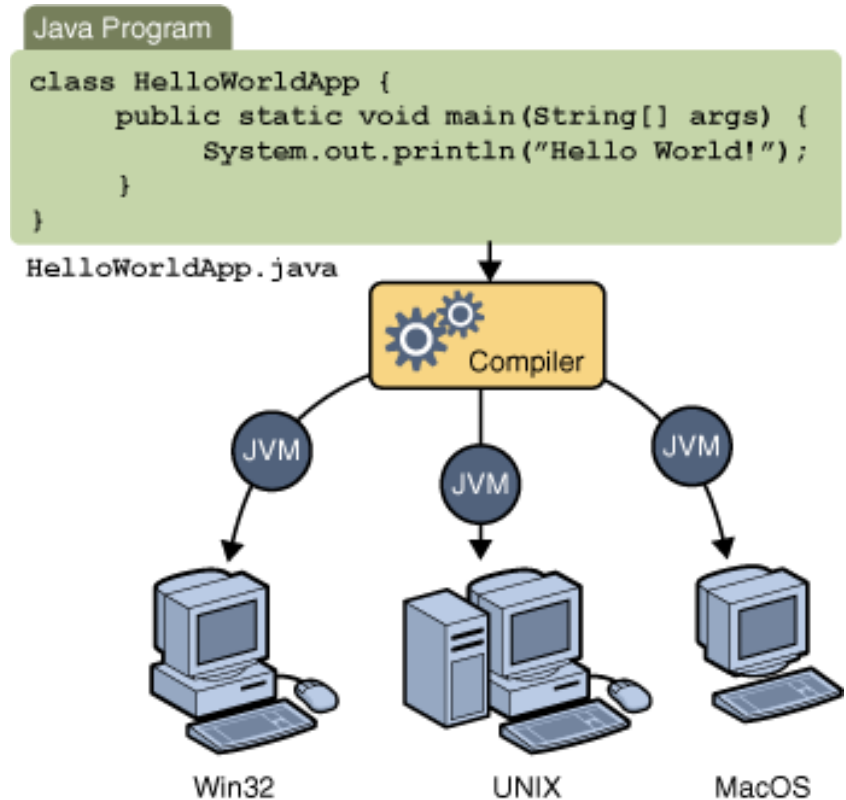
- Espacio en memoria que guarda un valor
- Tiene un “tiempo de vida” (*scope*)
- Tienen un nombre que debe ser único mientras se encuentre “viva”
- Una variable se
 - Declara
 - Inicializa
 - Utiliza



<http://img.c4learn.com/2012/02/Variable-in-Java.jpg>

Lenguaje Java

- Fue creado por Sun Microsystems en 1995
- Es **independiente** de la plataforma en donde se ejecute
- Es un lenguaje de **propósito general**
- Implementa el **paradigma de Orientación a Objetos**



<https://tecnoesis.files.wordpress.com/2009/08/java-cross-platform-support.gif?w=346&h=332>

Instrucciones en Java

- Una instrucción termina con un punto y coma
- Un bloque de instrucciones se agrupa en medio de corchetes
- Existen palabras reservadas

```
public class Programa 001
{
    public static void main(String[] args)
    {
        int a = 5;
        int b = 3;
        int c = a + b;
        System.out.println(c);
    }
}
```

Bloque de instrucciones externo

Bloque de instrucciones interno

instrucción

Sintaxis: Clases

- Cada clase se debe guardar en un archivo separado
- El nombre debe ser igual al nombre de la clase, y cuya extensión debe ser .java.
- Todo lo que va entre corchetes {...} se denomina un bloque de instrucciones.
- En Java las mayúsculas **SI IMPORTAN**

```
public class MiClase
{
    //Comentario: Aquí va la declaración de la clase
}
```

Sintaxis: comentarios

- Entre los corchetes de la clase va la declaración de la clase, en donde se hacen explícitos los atributos y los algoritmos.
- También se pueden introducir comentarios que le pueden ayudar al programador. Una de las formas de escribir comentarios es con doble slash (//) o (/**/)

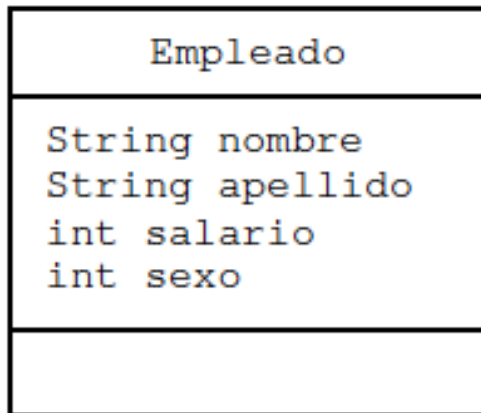
```
public class MiClase
{
    //Comentario: Aquí va la declaración de la clase
}
```


Sintaxis: Tipos de datos

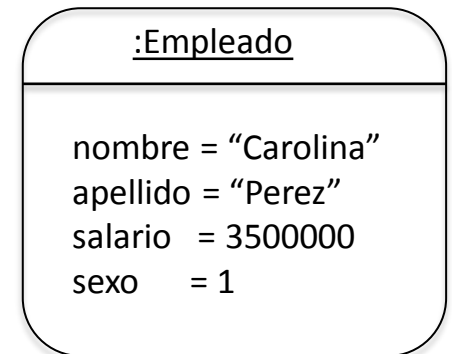
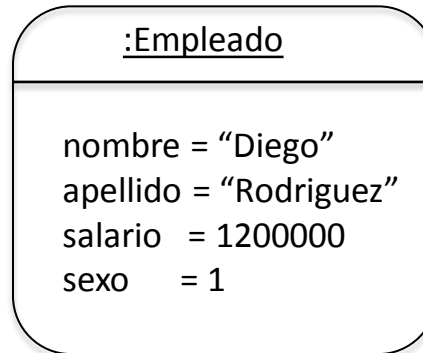
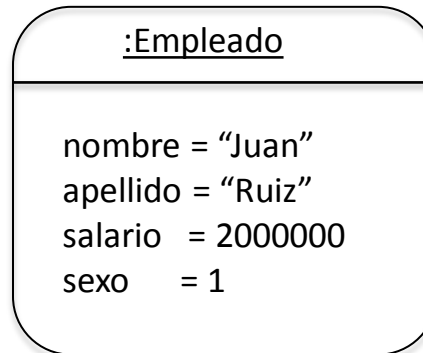
- Simples
 - Enteros (int)
 - Números enteros de -2.147.483.648 a 2.147.483.647
 - Reales (double)
 - Números reales de 2^{-1074} a 2^{1023}
 - Valores de verdad (boolean)
 - Verdadero o falso (true / false)
 - Cadenas de caracteres (String)
 - Palabras
- Definidos por el usuario

Clases y Objetos

Estructura



Código en Ejecución

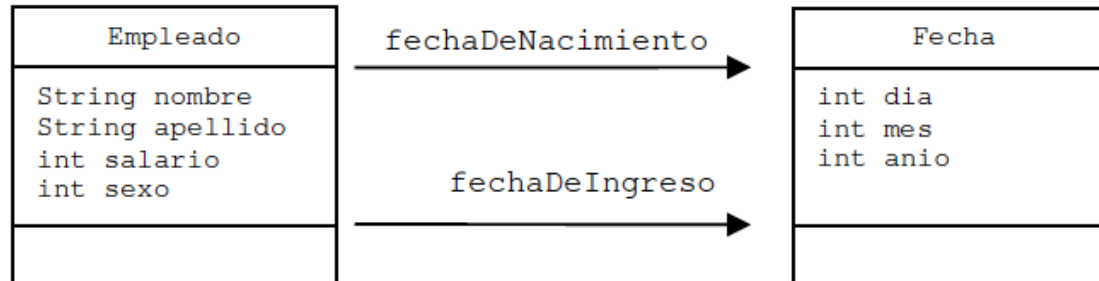


Estructura de una Clase

- Nombre
- Atributos
 - Variables de clase
 - Características de una clase
 - Tienen un tipo y un nombre único
- Métodos
 - Funcionalidad de la clase

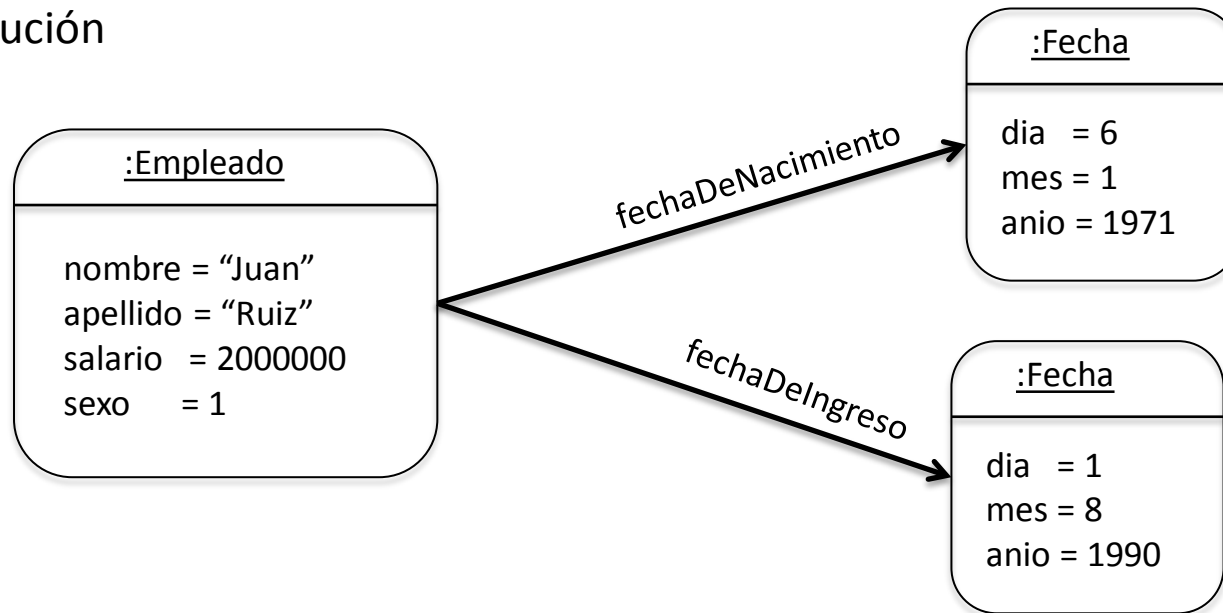
Empleado
<code>String nombre</code> <code>String apellido</code> <code>int salario</code> <code>int sexo</code>

Asociaciones



Estructura

Código en Ejecución



Práctica

- La Federación Colombiana de Ciclismo quiere tener un programa que le permita administrar los tiempos de los 4 ciclistas participantes en el tour de Francia. El usuario debe poder agregar el tiempo de una etapa a un ciclista, dar el tiempo total de un ciclista, y dar el promedio por etapa.

Estructura de una clase en Java

- Declaración de la clase
- Atributos
- métodos

```
/**
 * Esta clase representa un empleado
 */
public class Empleado {
    //
    // Atributos
    //-----
    private String nombre;
    private String apellido;
    private int salario;
    // 1 = masculino, 2 = femenino
    private int sexo;
    ....
}
```

Instrucciones: declaración de atributos

<visibilidad> **<tipo_de_dato>** **<nombre>;**

Empleado
String nombre String apellido int salario int sexo

```
package uniandes.cupi2.empleado;

/**
 * Esta clase representa un empleado
 */
public class Empleado {
    //-----
    // Atributos
    //-----
    private String nombre;
    private String apellido;
    private int salario;
    // 1 = masculino, 2 = femenino
    private int sexo;
    ....
}
```

Instrucciones: Asignación

- Se utiliza la instrucción de asignación para darle un valor a una variable.
- En Java, el símbolo de asignación es =
- El valor del lado derecho es asignado al lado izquierdo de la instrucción
- Ambos lados deben tener el mismo tipo de dato

salario = 200000;

nombre = "Juan";

Instrucciones: new

- Se utiliza la instrucción **new** para crear un objeto de una clase
- Un nuevo objeto se referencia (guarda) dentro de un atributo o variable del mismo tipo

Empleado e = **new** Empleado();

Fecha hoy = **new** Fecha();

¿PREGUNTAS?