

# Programación estructurada en Java

Area de Ingeniería Telemática  
<http://www.tlm.unavarra.es>

# Tipos de variables

# Tipos básicos simples

- Números enteros
  - Números punto flotante
  - Booleanos (condiciones)
  - Caracteres (no cadenas)
- 
- El nombre del tipo empieza con minúscula

```
public class Tipos {  
    public static void main(String[] args) {  
        int a;  
        double b;  
        boolean c;  
        char d;  
  
        a=3; a=42;  
        b=8.75; b=0.153;  
        c=true; c=false;  
        d='a'; d='x';  
  
        System.out.printf("a vale %d \n",a);  
        System.out.printf("b vale %f \n",b);  
        System.out.printf("c vale %b \n",c);  
        System.out.printf("d vale %c \n",d);  
  
    }  
}
```

# Tipos objetos

- El nombre del tipo empieza con Mayúscula
- Hay que usar new para construir un valor
- Hay operaciones sobre la variable
  - Por ejemplo en los de tipo String (cadenas)
  - longitud
  - comparar

```
public class Objetos {
    public static void main(String[] args) {
        String a;
        int l;

        a = new String("hola");
        l = a.length();

        System.out.printf("La cadena [ %s ] tiene longitud %d \n",a,l);

        if ( a.equals("hello") ) {
            System.out.println("La palabra es hello");
        } else if ( a.equals("hola") ) {
            System.out.println("La palabra es hola");
        } else {
            System.out.println("Palabra desconocida");
        }
    }
}
```

# Arrays

# Arrays

- Colección de valores del mismo tipo
- Tipo[] variable[4]
- El array en realidad es un objeto (operación length)
- new int[3] es de 0 a 2
- Acceder fuera da error

```
int[] a;
```

```
a = new int[3];
```

```
a[0]=3;
```

```
a[1]=27;
```

```
a[2]=58;
```

```
System.out.println("a[0] es "+a[0]);
```

```
System.out.println("a[1] es "+a[1]);
```

```
System.out.println("a[2] es "+a[2]);
```

```
String[] b;
```

```
b = new String[3];
```

```
b[0]="Alice";
```

```
b[1]="Bob";
```

```
b[2]="Charlie";
```

```
System.out.println("b[0] es "+b[0]);
```

```
System.out.println("b[1] es "+b[1]);
```

```
System.out.println("b[2] es "+b[2]);
```

# Recorrer un array

- Los argumentos del main son un array
- ¿Que hace este programa?

```
public class Args {
    public static void main(String[] args) {
        int i, n;

        n = args.length;

        i=0;
        while ( i<n ) {
            System.out.println("El argumento "+i+" es "+ args[i] );
            i=i+1;
        }
    }
}
```

- Ejercicio: hacer un programa que sume todos los argumentos que le pases (suponiendo que son números)

# Otro ejemplo

- Algunas funciones pueden devolvernos array
- Por ejemplo la operación sobre cadenas split corta las palabras y las devuelve en un array

```
public class Palabras {  
    public static void main(String[] args) {  
        String frase;  
        String[] palabras;  
        int i,n;  
  
        frase = new String("Cuantas palabras hay en una frase cualquiera");  
  
        palabras = frase.split(" ");  
        n=palabras.length;  
  
        System.out.println("En la frase hay "+n+" palabras");  
        i=0;  
        while ( i<n ) {  
            System.out.println("palabra "+i+" : "+palabras[i]);  
            i=i+1;  
        }  
    }  
}
```

- Ejercicio: hacer que cuente cuantas veces aparece una palabra concreta