

Práctica No. 2. Tipos de Datos y Entrada/Salida Estándar

Comenzar creando un nuevo proyecto tipo Java llamada Practica02, crear también la clase principal.

En el método **main**, se declararán varios de los tipos de datos que maneja Java y con el método de **System.out.println** se imprimirán.

Principal.java

```
package practica02;

import java.util.Scanner;

public class Principal {

    public static void main(String[] args) {

        Scanner lectura = new Scanner(System.in);

        int datoEntero = 5;
        float datoFlotante = 45.6F;
        double datoDoble = 145.6;
        boolean datoBooleano = true;
        char datoCaracter = 'a';

        System.out.println("Un entero: " + datoEntero);
        System.out.println("Un flotante: " + datoFlotante);
        System.out.println("Un doble: " + datoDoble);
        System.out.println("Un booleano: " + datoBooleano);
        System.out.println("Un caracter: " + datoCaracter);

    }

}
```

A continuación se leerán datos con la clase Scanner.

Se declarará un objeto de tipo **Scanner**

```
Scanner lectura = new Scanner(System.in);
```

Se debe importar la clase **Scanner**

```
import java.util.Scanner;
```

Se escribirá un mensaje para pedir un dato y se leerá.

Principal.java

```
package practica02;

import java.util.Scanner;

public class Principal {

    public static void main(String[] args) {

        Scanner lectura = new Scanner(System.in);

        int datoEntero = 5;
        float datoFlotante = 45.6F;
        double datoDoble = 145.6;
        boolean datoBooleano = true;
        char datoCaracter = 'a';

        System.out.println("Un entero: " + datoEntero);
        System.out.println("Un flotante: " + datoFlotante);
        System.out.println("Un doble: " + datoDoble);
        System.out.println("Un booleano: " + datoBooleano);
        System.out.println("Un caracter: " + datoCaracter);

        System.out.print("Ingresa un Entero:");
        datoEntero = lectura.nextInt();

        System.out.print("Ingresa un Flotante:");
        datoFlotante = lectura.nextFloat();

        System.out.print("Ingresa un Double: ");
        datoDoble = lectura.nextDouble();

        System.out.print("Ingresa un booleano: ");
        datoBooleano = lectura.nextBoolean();

        System.out.println("Un entero: " + datoEntero);
        System.out.println("Un flotante: " + datoFlotante);
        System.out.println("Un doble: " + datoDoble);
        System.out.println("Un booleano: " + datoBooleano);

    }
}
```

No es posible leer un carácter, para hacer esto, se puede leer como una cadena y posteriormente convertirlo a carácter.

Manejo de Cadenas de Caracteres

Para declarar una cadena de caracteres se utiliza la clase `String`, se puede declarar de varias maneras:

Principal.java

```
public static void manejoCadenas () {  
  
    String cadena = "";  
    String cadenaCaracteres = new String ();  
    String cadenaValor = new String ("Valor de la cadena");  
  
    System.out.println ("Cadena: " + cadena);  
    System.out.println ("Cadena: " + cadenaCaracteres);  
    System.out.println ("Cadena: " + cadenaValor);  
  
}
```

Lectura de Cadenas

Para leer una cadena, también se utiliza la clase `Scanner`, se tienen los métodos `nextLine()` y `next()`, la diferencia es que `next()` no lee espacios, mientras que `nextLine()` si los lee.

Principal.java

```
public static void manejoCadenas () {  
  
    String cadena = "";  
    String cadenaCaracteres = new String ();  
    String cadenaValor = new String ("Valor de la cadena");  
  
    Scanner lectura = new Scanner (System.in);  
  
    System.out.print ("Leyendo una cadena con nextLine: ");  
    cadena = lectura.nextLine ();  
  
    System.out.print ("Leyendo una cadena con next (No lee espacios): ");  
    cadenaCaracteres = lectura.next ();  
  
    System.out.println ("Cadena: " + cadena);  
    System.out.println ("Cadena: " + cadenaCaracteres);  
    System.out.println ("Cadena: " + cadenaValor);  
  
}
```

Caracteres y Cadenas

A continuación se lee una cadena y se convierte a un carácter (tomando el primer elemento)

Principal.java

```
public static void manejoCadenas () {

    String cadena = "";
    String cadenaCaracteres = new String ();
    String cadenaValor = new String ("Valor de la cadena");
    String cadenaCaracter;
    char caracter;

    Scanner lectura = new Scanner (System.in);

    System.out.print ("Leyendo una cadena con nextLine: ");
    cadena = lectura.nextLine ();

    System.out.print ("Leyendo una cadena con next (No lee espacios): ");
    cadenaCaracteres = lectura.next ();

    System.out.println ("Cadena: " + cadena);
    System.out.println ("Cadena: " + cadenaCaracteres);
    System.out.println ("Cadena: " + cadenaValor);

    System.out.print ("Ingresa un caracter:");
    cadenaCaracter = lectura.next ();
    caracter = cadenaCaracter.charAt (0);
    System.out.println ("El caracter es: " + caracter);

}
```

Operaciones con Cadenas

Se presentan las operaciones más comunes que se pueden realizar con cadenas de caracteres.

Principal.java

```
public static void operacionesCadenas () {

    String cadena;
    int posicion;
    int comparando;
    Scanner lectura = new Scanner (System.in);

    System.out.print ("Ingresa una cadena: ");
    cadena = lectura.nextLine ();

    /*Longitud y concatenación*/
    System.out.println ("Longitud: " + cadena.length ());
    System.out.println ("Concatenando: " + cadena.concat ( " Hola! "));
    //cadena = cadena.concat (" Hola!");
    System.out.println ("Longitud: " + cadena.length ());

}
```

```
/*Comparación de cadenas*/
comparando = cadena.compareTo("Zebra");
System.out.println("Comparación: " + comparando);

comparando = cadena.compareTo("Hola");
System.out.println("Comparación: " + comparando);

comparando = cadena.compareTo("Ave");
System.out.println("Comparación: " + comparando);

/*Búsqueda de palabras*/
posicion = cadena.indexOf("como");
System.out.println("Posición: " + posicion);

posicion = cadena.indexOf("perro");
System.out.println("Posición: " + posicion);

}
```