Javascript :El segundo es un lenguaje de scripting que permite agregar cierta funcionalidad dinámica en las páginas Web. La tecnología Java que permite construir este tipo de aplicaciones está basada en el desarrollo de Servlets, pero esto es parte de lo que se conoce como JEE (Java Enterprise Edition) y

package libro.cap01;

public class HolaMundo

{

public static void main(String[] args)

{

System.out.println("Hola Mundo !!!");

}

}

Bloques-sentencias

Un programa Java es una clase (class) que contiene el método (o función) main

Los bloques de código se delimitan con { } (llaves) y las sentencias finalizan con ; (punto y coma).

En Java siempre codificamos clases y cada clase debe estar contenida dentro de un archivo de texto con el mismo nombre que la clase y con extensión .java. Así, nuestro programa debe estar codificado en un archivo llamado HolaMundo.java (respetando mayúsculas y minúsculas).

1.2.1 El Entorno Integrado de Desarrollo (IDE) Si bien podemos editar nuestro código utilizando cualquier editor de texto y luego compilarlo en línea de comandos, lo recomendable es utilizar una herramienta que nos ayude en todo el proceso de programación. Una IDE (Integrated Development Enviroment) es una herramienta que permite editar programas, compilarlos, depurarlos, documentarlos, ejecutarlos, etc. (eclipse – netbeans)

1.3.1 Entrada y salida de datos por consola Llamamos “consola” al conjunto compuesto por la pantalla (en modo texto) y el teclado de la computadora donde se ejecutará nuestro programa.

package libro.cap01;

import java.util.Scanner;

public class HolaMundoNombre

{

public static void main(String[] args)

{

// esta clase permite leer datos por teclado

Scanner scanner = new Scanner(System.in);

// mensaje para el usuario

System.out.print("Ingrese su nombre: ");

// leemos un valor entero por teclado y lo almaceno en una variable

String nom = scanner.nextLine();

// mostramos un mensaje y luego el valor leido

System.out.println("Hola Mundo: " + nom);

}

}

La clase Scanner permite leer datos a través del teclado

Diferencia entre: System.out.print y System.out.println. El primero imprime en la consola el valor del argumento que le pasamos. El segundo hace lo mismo pero, agrega un salto de línea al fi nal.

1.3.2 Defi nición de variables Podemos defi nir variables en cualquier parte del código simplemente indicando el tipo de datos y el nombre de la variable (identifi cador).

-no debe comenzar con numero o signo ( guion)

1.3.3 Comentarios en el código

Java soporta comentarios in-line (de una sola línea) y comentarios de varias líneas.

Comentarios de una sola línea:

// esto es una linea de codigo comentada

Comentarios de más de una línea:

/\*

Estas son varias

lineas de codigo

comentadas

\*/

\*\*argumento(parámetro)

package libro.cap01;

import java.util.Scanner;

public class HolaMundoNombre

{

public static void main(String[] args)

{

Scanner scanner = new Scanner(System.in);

// Mensaje para el usuario

System.out.print("Ingrese nombre edad altura: ");

// leemos el nombre

String nom = scanner.next();

// leemos la edad

int edad= scanner.nextInt();

// leemos la altura

double altura = scanner.nextDouble();

// mostramos los datos por consola

System.out.println("Nombre: "+nom

+" Edad: "+edad

+" Altura : "+altura);

}

}

1.3.4 Estructuras de decisión En Java disponemos de tres estructuras de decisión o estructuras condicionales: Decisión simple: if

Decisión múltiple: switch

Decisión in-line: a>b ? "a es Mayor" : "a es Menor" /String mssg = (resto == 0 ) ? "es Par": "es Impar";

Comenzaremos analizando la sentencia if cuya estructura es la siguiente:

if( condicion )

{

accion1;

}

else

{

accion2;

}

Java (como C) utiliza == como operador de comparación ya que el operador = (igual) se utiliza para la asignación.

La decisión múltiple switch tiene la siguiente estructura:

switch( variableEntera )

{

case valor1:

accionA;

accionB;

:

break;

case valor2:

accionX;

accionY;

:

break;

:

default:

masAcciones;

}

variableEntera debe coincidir con el valor Case , en este caso se ejecuta las instrucciones dentro del bloque y la sentencia break(sale de la estructura) , en caso de no coincidir el ejecutar el bloque default.