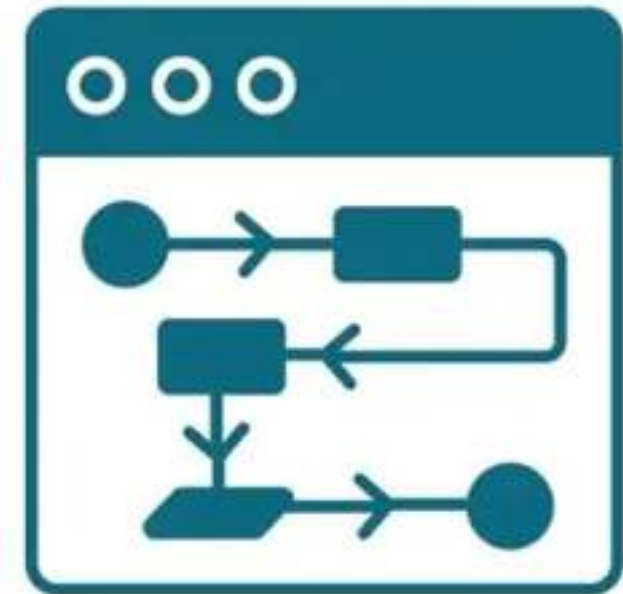




Algoritmos y Estructura de Datos

PROGRAMACIÓN EN JAVA



DOCENTE: MARX DANLY LEÓN TRUJILLO
FACULTAD DE ING. INDUSTRIAL Y SISTEMAS
E.P. INGENIERÍA DE SISTEMAS

PROGRAMACIÓN EN JAVA

- Introducción
- Declaraciones
- Estructura General
- Sentencias de Selección IF, IF-ELSE, SWITCH.
- Ejercicios.



PROGRAMACIÓN EN JAVA

HISTORIA

- ❖ Creado por Sun Microsystem Inc.
- ❖ Presentado y Distribuido en el año 1995
- ❖ Desarrollado como proyecto liderado por James Gosling.



JAVA

Java es un lenguaje de desarrollo de propósito general y como tal es válido para realizar todo tipo de aplicaciones profesionales.



ESTRUCTURA GENERAL DE UN PROGRAMA EN JAVA

Un programa puede construirse empleando varias clases. En el caso más simple se utilizará una única clase. Esta clase contiene el programa, rutina o método principal: **main()** y en éste se incluyen las sentencias del programa principal.

La estructura de un programa simple en Java es la siguiente:

```
public class ClasePrincipal {  
    public static void main(String[] args) {  
        sentencia_1;  
        sentencia_2;  
        // ...  
        sentencia_N;  
    }  
}
```

ESTRUCTURA GENERAL DE UN PROGRAMA EN JAVA

Ejemplo:

```
package javaapplication1;

/**
 *
 * @author HP
 */
public class JavaApplication1 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        String curso="ALGORTIMOS Y ESTRUCTURA DE DATOS";
        System.out.println("Bienvenidos al curso de '"+curso+"' ");
    }
}
```

ESTRUCTURA GENERAL DE UN PROGRAMA EN JAVA

Todos los programas o aplicaciones independientes escritas en Java tienen un método main o principal que, a su vez, contiene un conjunto de sentencias. En Java los conjuntos o bloques de sentencias se indican entre llaves { }.

```
int metodoEntero()//Función sin parámetros
{
    int suma = 5+5;
    return suma; //Acá termina la ejecución del método
    //return 5+5;//Este return nunca se ejecutará
    //Intenta intercambiar la línea 3 con la 5
    //int x = 10; //Esta línea nunca se ejecutará
}
```

DECLARACIONES

VARIABLES Y CONSTANTES

VARIABLES:

Podemos definir variables en cualquier parte del código simplemente indicando el tipo de datos y el nombre de la variable (identificador).

Identificadores válidos son:

```
fecha  
iFecha  
fechaNacimiento  
fecha nacimiento  
fecha3  
_fecha
```

Identificadores NO válidos son:

```
3fecha  
fecha-nacimiento  
fecha+nacimiento  
-fecha
```

DECLARACIONES

Variables de tipo primitivo

```
//Declaración de la variable  
tipo nombreVariable;  
  
//Se le asigna un valor COMPATIBLE a la variable  
nombreVariable = valor;  
  
//Se declara la variable y se le asigna un valor al mismo tiempo  
tipo nombreVariable = valor;
```


DECLARACIONES

Variables de tipo complejo o estructurado

```
//Declaración de la variable
TipoComplejo nombreVariable;

//Se le inicializa por medio del constructor la variable
nombreVariable = new TipoComplejo(parametro);

//Se declara la variable y se le asigna un valor al mismo tiempo
TipoComplejo nombreVariable = new TipoComplejo(parametro);
```

DECLARACIONES

Constantes

La palabra clave que java ha reservado para definir constantes es la palabra "final". En java es muy simple definir constantes, solo basta con ponerles el modificador final antes de la declaración del tipo.

```
final tipo nombreVble = valor;
```

```
public class Constantes
{
    private final String atribConstante;
    //Notar que no ha sido inicializado

    //Constructor
    Constantes(String valor)
    {
        //Aquí se inicializó el atributo
        atribConstante = valor;
    }
}
```

DECLARACIONES

Arreglos y Matrices

Un array es una estructura de datos que nos permite almacenar una gran cantidad de datos de un mismo tipo. El tamaño de los arrays se declara en un primer momento y no puede cambiar en tiempo de ejecución como puede producirse en otros lenguajes. Lo mismo se cumple con las matrices en Java.

```
public class ArreglosYMatrices2 {  
    public static void main(String args[]) {  
        int[] arreglo = {1,2,3,4,5,6};  
        int arreglo2[] = {1,2,3,4,5,6};  
        int[][] matriz = {{10,11,12,13},{20,21,22},{30,31},{4},{5},{6}};  
        int matriz2[][] = {{10,11,12,13},{20,21,22},{30,31},{4},{5},{6}};  
    }  
}
```

SENTENCIAS DE SELECCIÓN O CONDICIONALES

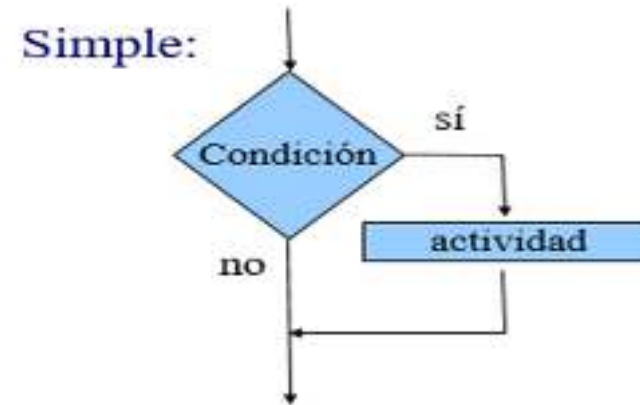
Se usan para la toma de decisiones dentro de un programa y lo podemos clasificar:

1. Simple → (Si_entonces) **(IF)**
2. Doble → (Si_entonces_sino) **(IF_ELSE)**
3. Múltiple (según_sea) → (Si_múltiple) **(SWITCH)**

SENTENCIAS DE SELECCIÓN O CONDICIONALES

SENTENCIA IF

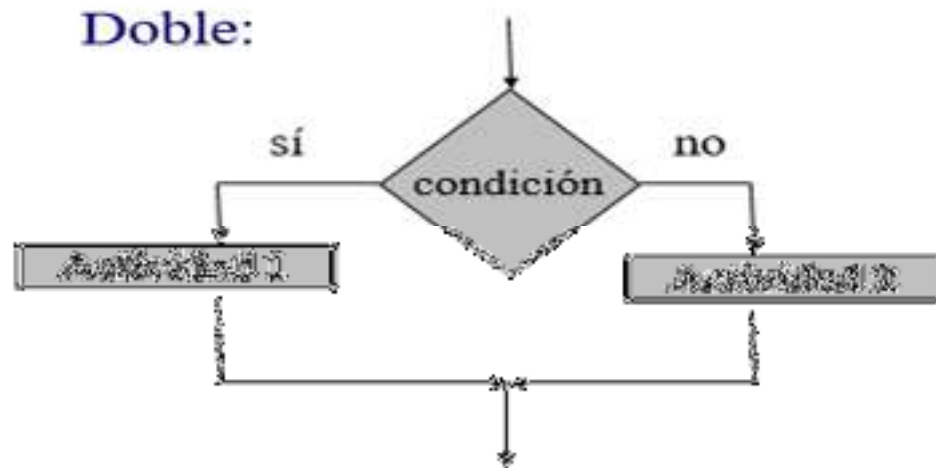
Permite ejecutar una acción o grupo de acciones solo si se cumple una determinada condición, es decir, donde la condición es un grupo de condiciones a evaluarse. El bloque “A” representa la operación o conjunto de operaciones a realizar si la condición es verdadera.



```
if( condicion )  
{  
  accion1;  
}
```

SENTENCIAS DE SELECCIÓN O CONDICIONALES

SENTENCIA IF_ELSE



```
if( condicion )  
{  
  accion1;  
}  
else  
{  
  accion2;  
}
```

SENTENCIAS DE SELECCIÓN O CONDICIONALES

SENTENCIA SWITCH

PSEUDOCODIGO	EN JAVA
Según sea(Valor) Caso <condicion1> Sentencia1(s) Salto Caso <condicion2> Sentencia2(s) Salto Caso <condicion3> Sentencia3(s) Salto Caso <condicion4> Sentencia4(s) Salto Default SentenciaD(s) Salto FinSegunsea	switch (valor) { Case <condicion1>: Sentencia1(s); break ; Case <condicion2>: Sentencia2(s); break ; Case <condicion3>: Sentencia3(s); break ; Case <condicion4>: Sentencia4(s); break ; default : SentenciaD(s); break ; }

SENTENCIAS DE SELECCIÓN O CONDICIONALES

Ejercicio

Escribir un algoritmo que pida el colegio de Procedencia que puede ser A, B, C y el modo de educación tipo Nacional (N), y Particular (P). Según el siguiente cuadro hallar el monto a pagar.

	Nacional	Particular
A	50	25
B	40	29
C	30	15



SENTENCIAS DE SELECCIÓN O CONDICIONALES

Ejercicio

Comprobar si un carácter es un dígito, una letra mayúscula o una letra minúscula

Código ASCII de:

- dígito: 48 – 57
 - letra mayúscula: 65 – 90
 - letra minúscula: 97 – 122
- 