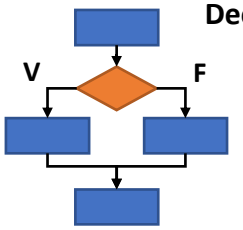
Conceptos de Programación (en Java)

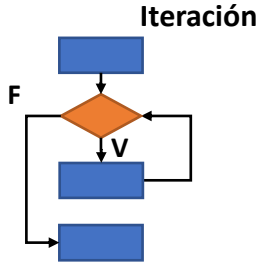
Sabemos que la memoria es frágil, así que te entregamos esta pequeña guía de ayuda...

Parte 1: Java básico

Algoritmo

Secuencia de pasos ordenados y finitos, cuya ejecución permite resolver un problema o necesidad





Estructuras de control

Estructuras básicas de programación que definen la secuencia y orden con que se ejecuta un algoritmo (*if-else*, *while*, etc.)

Declaración de variable

Expresión que permite declarar una variable a utilizar, indicando su tipo. Toda variable debe ser declarada para ser usada

```
int dato = 10;
```

Inicialización de variable

Expresión que asigna el primer valor a una variable, usando el carácter '='. Toda variable debe ser inicializada antes de ser usada

```
for(int i=0; i < 10; i++){
    int d = 2; d++;
    if(i + d > 10)
        System.out.println("Mayor a 7");
}
```

Bloque

Secuencia de instrucciones agrupadas comúnmente dentro de una estructura de control. Toda variable declarada dentro de un bloque no puede ser usada fuera del mismo

Tipos primitivos

Tipos de datos elementales, que forman de base para guardar datos en los programas (enteros, decimales, caracteres, etc.)

int byte char double boolean

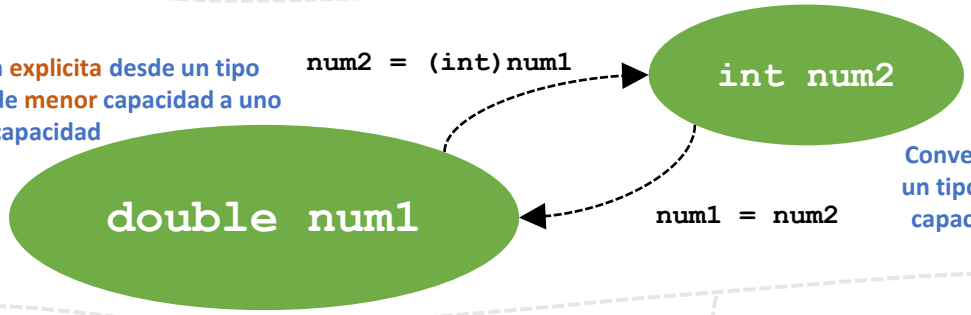
Clases

Tipos de datos avanzados, formados de primitivos, que permiten manejar conceptos complejos

BufferedReader IOException String CuentaAhorro Persona

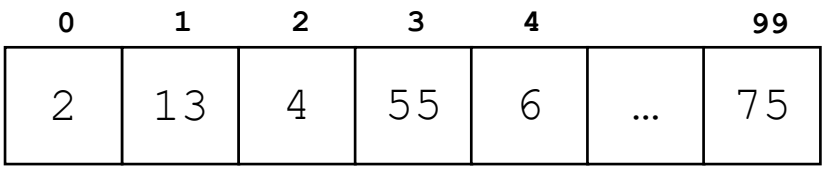
Casting

Conversión explicita desde un tipo numérico de menor capacidad a uno de mayor capacidad



Promoción automática

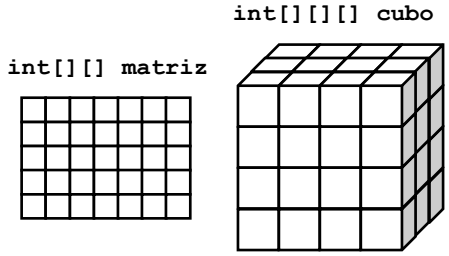
Conversión implícita desde un tipo numérico de mayor capacidad a uno de menor capacidad




```
int[] datos = new int[100]
```

Arreglo simple

Estructura de datos básica en Java. Corresponde a un vector de largo finito, especificado en su creación, en donde cada posición se manipula con un índice y responde exactamente al mismo tipo de dato con que el arreglo fue declarado



Arreglo multidimensional

Extensión de un arreglo simple, capaz de manipular más de una dimensión (índice)

Enrique Urra (enrique.urra@uaysen.cl), 2023

Conceptos de Programación (en Java)

Sabemos que la memoria es frágil, así que te entregamos esta pequeña guía de ayuda...

Parte 2: Orientación a Objetos (OO)

Clase

Tipo de dato avanzado, definido en el diseño de una aplicación, que permite especificar los datos y comportamientos que puede tener la entidad que representa

Instanciar

Crear un objeto desde la definición de la clase, usando el operador `new`

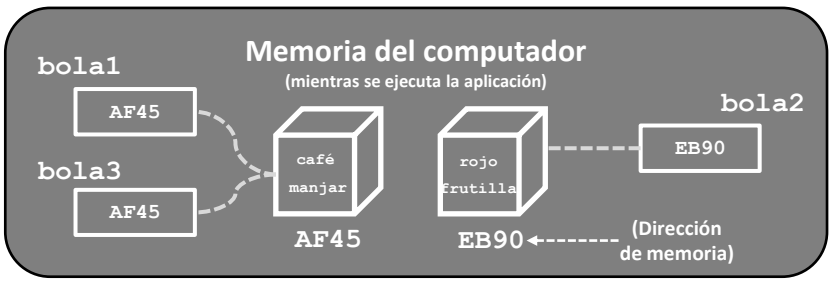
Objeto

Instancia de una clase que se crea mientras se ejecuta la aplicación, que guarda valores de los datos cambiantes en el tiempo, y en base a los cuales el comportamiento de la entidad opera

```
public class BolaHelado {
  private String color;
  private String sabor;

  public BolaHelado(String c,
                    String s) {
    color = c;
    sabor = s;
  }

  public String comer() {
    return "Delicioso helado
           de " + sabor + "!";
  }
}
```




```
BolaHelado bola1 = new BolaHelado("café","manjar");
BolaHelado bola2 = new BolaHelado("rojo","frutilla");
BolaHelado bola3 = bola1;
```

Constructor

Operación especial que se invoca al instanciar un objeto con el operador `new`

Variable de referencia

Variable cuyo tipo de dato corresponde a una clase. Funciona como un puntero a la ubicación de un objeto en memoria, por ello, más de una variable de referencia puede referenciar a un mismo objeto

Atributo o variable de instancia

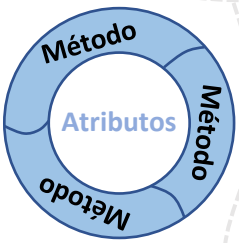
Implementación de alguno de los datos de una clase

Método

Implementación de alguno de los comportamientos de una clase

Encapsulamiento

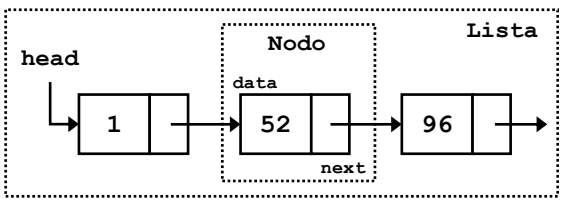
Capacidad de un objeto para ocultar los detalles de su implementación (atributos) a través de una interfaz bien definida (métodos y constructores)



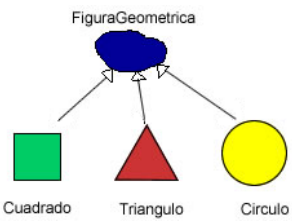
Polimorfismo

Capacidad de varios objetos para comportarse de manera específica o diversa, a través de una interfaz común (métodos)


```
animal.hablar();
```



Herencia



Capacidad de un subtipo (una subclase) para extender un supertipo (super clase) y reutilizar y/o redefinir su implementación. Siendo X el subtipo e Y el supertipo, la herencia es una relación descrita como *X es un tipo de Y*

Lista simplemente enlazada

Estructura de datos que guarda una cantidad indeterminada de elementos, enlazados unos con otros y de manera secuencial a través de nodos, siendo la cabeza (`head`, el primer nodo) el único punto de acceso a ellos

Enrique Urra (enrique.urra@uaysen.cl), 2023