Especificadores de formato (printf)

Comienzan con un signo de porcentaje (%) y van seguidos de un carácter que representa el tipo de datos. Por ejemplo, el especificador de formato **%s** es un receptáculo para una cadena. Esto quiere decir que despliega una segunda cadena de caracteres en la línea siguiente. Cabe de destacar que en vez de usar la secuencia de escape \n, usamos el especificador de formato %n para separar líneas.

Otro especificador de formato es el %d  que es un receptáculo para un valor **int**; la letra **d** se refiere a entero decimal

Declaraciones import

    Una gran fortaleza de Java es su extenso conjunto de clases predefinidas que podemos reutilizar. Estas clases se agrupan en **paquetes** y se conocen como **biblioteca de clases de Java o Interfaz de programación de aplicaciones de Java (API de Java).**

**Nota**: Todas las declaraciones de **import** deben aparecer antes de la primera clase en el archivo, si no, puede causar error de sintaxis. Si se olvida importar, puede aparecer error de compilación.

    Un ejemplo de estas clases puede ser la clase **Suma**. El nombre del archivo para esta clase public debe ser Suma.java

    Existen paquetes que se importan automáticamente como en el caso de java.lang. Son las únicas en el API de java que no requieren declaración de import

**Tipos de Variables**

    Una variable es una ubicación en la memoria de la computadora, donde se puede guardar y acceder a un valor para poder utilizarse en cualquier momento. Existe en java distintos tipos de variables:

|  |  |
| --- | --- |
| int | Guardar números enteros |
| float y double | Guardar números reales |
| char | Guardar datos de caracteres individuales |
|  |  |

    A estos se le conoce como de tipo primitivo. Es posible declarar varias variables de una misma clase con sólo una declaración, separando por comas:

    int numero1, numero2, suma;

    Los identificadores de nombres de las variables empiezan con minúscula por convención

**Obtener Valores de entrada de usuario**

    Se utiliza el método **.nextInt()** del objeto entrada de la clase Scanner para obtener un entero del usuario mediante el teclado. El programa espera a que el usuario escriba el número y oprima la tecla Enter para enviar el dato al programa. Siempre se debe hacer validación de los datos ingresados por el usuario.

**Operaciones aritméticas**

    La suma, resta, multiplicación y división se ejecuta primero las operaciones de mayor peso y luego las de menor peso y de izquierda a derecha, todo de manera lineal para ayudar a la creación del código.

**Operaciones de toma decisiones: igualdad y relaciones**

    == representa igual que

    ¡= representa no igual que

    > mayor que

    < menor que

    >= mayor o igual que

    <= menor o igual que

    Estas **condiciones** son expresiones de verdadero o falso. Se presenta con la instrucción **if** de java, la cual permite tomar decisiones.

**Instrucción if**

Para realizar una condicional con la instrucción **if;** se coloca la instrucción, seguida por la condición entre paréntesis y luego la operación que se realiza si se cumple la condición y cerrando con el punto y coma:

if (numero1 == numero2)

System.out.printf(“%d == %d%n”, numero1, numero2);