



Agencia de
Aprendizaje
a lo largo
de la vida

Codo a Codo inicial

Clase 6

Les damos la bienvenida

Vamos a comenzar a grabar la clase

Clase 05

Clase 06

Clase 07

Sintaxis y variables

- **Variables.** Características, declaración, asignación, semántica, alcance.
- **Relación entre variables y datos.**
- **Constantes.**
- **Tipado.**
- **Desafíos.**

Operadores de asignación - Operadores aritméticos - input y output

- Repaso General.
- Definición de variables.
- Asignación de valores.
- Asignación de resultados.
- E/S de valores por pantalla.

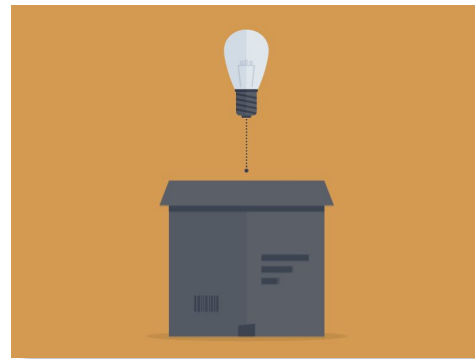
Operadores relacionales

- Operadores relacionales
- Distintos operadores.
- Ejemplos de aplicación.

En el capítulo anterior...

Variables

- **Una variable** en programación es utilizada de forma habitual.
- Se trata de un elemento fundamental para **gestionar la información** que se decida incorporar en el código y básicamente **funciona para guardar datos**.
- **Una variable almacena un valor único.**



Características de una variable en programación

1. Una variable **reserva espacio de memoria**.
2. Una variable debe ser **declarada para que exista**.
3. Una variable tiene un **alcance global** es decir puede ser utilizada en todo el proyecto o **local**, solo en las pocas subsiguientes líneas de código de un bloque en particular.

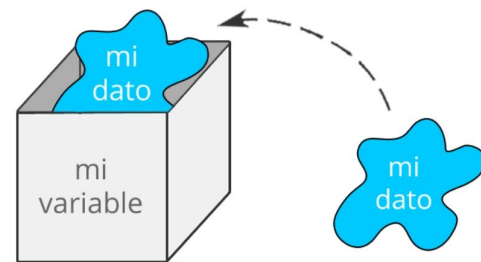


Variable y asignación de valores (repaso)

Declaración e inicialización de una variable

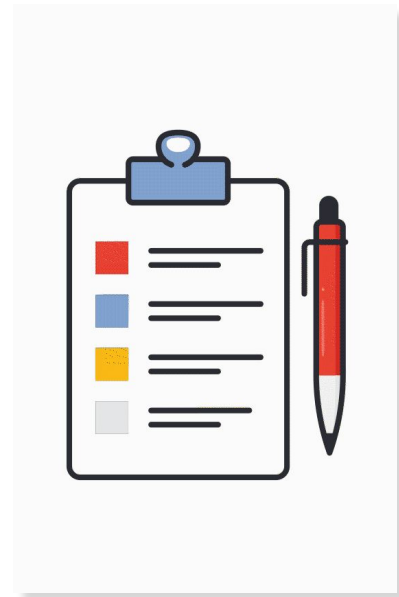
- **Una variable se declara** para indicarle al programa **a partir de qué lugar empieza a existir**, qué **nombre** tendrá y qué tipo de **datos almacenará**.
- Para declarar una variable usaremos una instrucción compuesta del nombre del **tipo de datos de la variable**, el **nombre de la variable**
- La variable se asigna o inicializa **con un operador de asignación (signo =) y un valor inicial**.

Ejemplo miNumero = 5



Buenas prácticas

- Como buena práctica te sugiero **declarar todas las variables al principio del programa**, por más que haya alguna que vayas a utilizar a lo último.
- Esto permite una **mejor legibilidad del programa**.



Tipos de Variables

Esquema de los tipos de datos



Operadores de asignación

Inicialización de variables

Operador de asignación

- Hasta ahora las variables **no están inicializadas**, es decir, **su valor es indefinido**.
- Para iniciarlas en un valor, se utiliza el **operador de asignación** de **PSelnt** es el símbolo =



Fuente: C. Cimino

Sintaxis de la asignación a una variable

La sintaxis para asignar un valor a una variable es la siguiente:

`<nombre_de_la_variable> = <expresión>;`

- La palabra **<nombre_de_la_variable>** indica que en ese lugar debe ir el identificador de una variable definida.
- **=** es el operador de asignación.
- La palabra **<expresión>** indica que en ese lugar debe ir una expresión válida, de un **tipo de dato compatible** con lo que espera guardar la variable.
- El **;** en algunos lenguajes es obligatorio e indica que finalizó la instrucción.

PseInt. Asignando valores a variables

```
1 //Operador de asignación
2 //Clase 6 CaC Inicial
3
4 Algoritmo carga_muestra_variables
5     Definir edad Como Entero;
6     Definir sueldo Como Real;
7     Definir hijos Como Entero;
8     Definir nombre Como Cadena;
9     nombre = "Carlos";
10    hijos = 0;
11    sueldo = 12345.67;
12    edad = 25;
13    Escribir sueldo;
14    Escribir nombre;
15    Escribir edad;
16    Escribir hijos;
17 FinAlgoritmo
18
```

Instrucciones de entrada

Operadores aritméticos

Operadores de concatenación

Instrucción de entrada

- Una **instrucción de entrada** nos permite interactuar con la aplicación.
- Permite **recibir datos introducidos por el usuario**.
- Estos datos serán almacenados en una variable
- Posteriormente estos datos serán procesados para producir un resultado

PSInt - Ejecutando proceso INSTRUCCIONENTRADA

*** Ejecución Iniciada. ***

Ingresá tu nombre

> |

línea 3 instrucción 1

Instrucción de entrada

La sintaxis **en PseInt** es la siguiente:

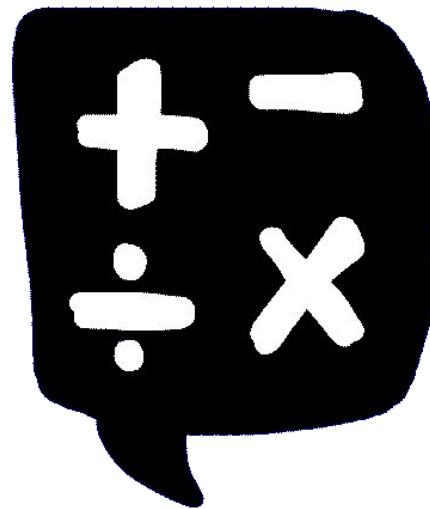
Leer <nombre_de_la_variable>;

- **Leer** se tipea literalmente, indica que **queremos obtener un dato** desde el **teclado**.
- La palabra <nombre_de_la_variable> indica en qué variable se almacenará el dato ingresado **por el usuario vía teclado**.
- El **;** es obligatorio en algunos lenguajes e indica que finalizó la instrucción.



Operadores aritméticos

Consta de los operadores fundamentales de la aritmética con el agregado de la potencia y el módulo o residuo, que devuelve el resto entero que se produce al realizar un cociente entre dos números enteros.



Carlos E. Cimino

Operador	Nombre	Ejemplo	Resultado	Descripción
+	Suma	$12 + 3$	15	Devuelve la suma de dos expresiones.
-	Resta	$12 - 3$	9	Devuelve la resta de dos expresiones.
*	Multiplicación	$12 * 3$	36	Devuelve la resta de dos expresiones.
/	División	$12 / 3$	4	Devuelve el cociente de dos expresiones.
^	Potenciación	$12 ^ 3$	1728	Devuelve la potencia entre dos expresiones.
%	Módulo o Residuo	$12 \% 3$	0	Devuelve el resto del cociente entre dos enteros.

PseInt. Instrucciones de entrada.

```
1  //Aplicación con instrucciones de entrada
2  //CaC Inicial
3
4  Algoritmo sumador
5      Definir numero1 Como Entero;
6      Definir numero2 Como Entero;
7      Escribir "SUMADOR";
8      Escribir "Ingrese el primer número entero:";
9      Leer numero1;
10     Escribir "Ingrese el segundo número entero:";
11     Leer numero2;
12     Escribir "El resultado es";
13     Escribir numero1 + numero2;
14 FinAlgoritmo
```

Operador de concatenación

- Para hacer más atractivas las salidas, se utiliza el operador de concatenación, que **es la coma**.
- Este permite unir cadenas de textos unas con otras o con contenidos de variables **para una mejor lectura en la salida de datos**.



PseInt. Concatenadores.

```
1  //Operador de concatenacion
2  //CaC Inicial
3
4  Algoritmo sumador_salida_concatenada
5      Definir numero1 Como Entero;
6      Definir numero2 Como Entero;
7      Escribir "SUMADOR";
8      Escribir "Ingrese el primer número entero:";
9      Leer numero1;
10     Escribir "Ingrese el segundo número entero:";
11     Leer numero2;
12     Escribir "El resultado es " , (numero1 + numero2);
13 FinAlgoritmo
14
```

Constantes

- **Son elementos de almacenamiento de datos.**
- Representan una dirección de memoria en donde se almacena un dato **pero que no varía durante la ejecución del programa.**
- Ejemplo, el número $\pi = 3.14$.



Desafíos

Desafío de clase

Realizar un programa que:

- Pida al usuario que ingrese su nombre
- Pida su fecha de nacimiento

Luego de ingresar estos datos el usuario recibirá un saludo personalizado y podrá visualizar su edad.

Desafío de clase - Resolución

Algoritmo Calculadora_de_edad

// Declaramos las variables

Definir nombre **Como** Caracter;

Definir anioAct **Como** Entero; *// Aca va el año actual*

Definir anioNac **Como** Entero; *// Aca va el año de nacimiento*

Definir resultadoEdad **Como** Entero;

// Inicializacion de variables E/S

Escribir "Por favor, ingresá tu nombre: ";

Leer nombre;

Escribir "¿Podrías decirme que año es actualmente?: "

Leer anioAct;

Escribir "¿Podrías introducir el año en que naciste?: "

Leer anioNac;

// Procesos

resultadoEdad = (anioAct-anioNac);

// Resultado

Escribir "Hola! ", nombre;

Escribir "Naciste en el año ", anioNac, " y tenés ", resultadoEdad, " años, si es que ya los cumpliste."

Escribir "*** El programa ha finalizado satisfactoriamente ***"

FinAlgoritmo

Desafío II de clase

Codificar una calculadora de multiplicación y restas entre dos números enteros para obtener su resultado por consola (pantalla).

Desafío III de clase

1. Pedir las horas trabajadas de una persona y el valor por hora para calcular su salario e imprimirlo por pantalla.

Desafío V sobre nuestro algoritmo

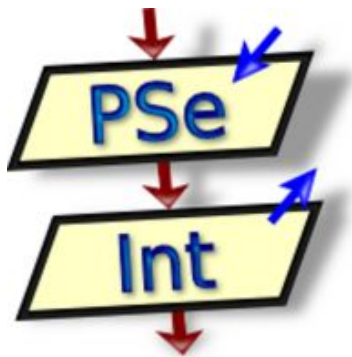
Seguimos integrando contenido y enriqueciendo nuestro algoritmo inicial.

Se pide:

- Enriquecer el código con instrucciones de Entrada y Salida y operadores aritméticos y concatenadores.
- Refactorizá tu código con estas nuevas determinaciones.



Herramientas que utilizamos en clases



PSeInt es una herramienta para asistir a un estudiante en sus primeros pasos en programación

No te olvides de dar el presente

Recordá:

- **Revisar la Cartelera de Novedades.**
- **Hacer tus consultas en el Foro.**
- **Realizá los ejercicios obligatorios.**

Todo en el Aula Virtual.