



Recibe una cálida:

¡Bienvenida!

Te estábamos esperando 😊 

» Algoritmos

Plan formativo: Desarrollo de Aplicaciones Full Stack Java Trainee V2.0

HOJA DE RUTA

¿Cuáles **skill** conforman el programa?



LEARNING PATHWAY

4.2

Start! 🚩

Fundamentos de programación en Java

Introducción a algoritmos.
Características y componentes.
Variables y tipos de datos.

Algoritmos: concepto y características

Variables y tipos de datos

Actividad N° 1: Creando un algoritmo

OBJETIVOS DE APRENDIZAJE

¿Qué aprenderemos?



Comprender el concepto y las características de los algoritmos



**Entender el concepto y las características de las variables.
Uso e implementación.**



Conocer los Tipos de Datos y las operaciones que pueden realizarse con cada uno.



REPASO CLASE ANTERIOR

En la clase anterior trabajamos :

- ✓ El lenguaje Java y sus características.
- ✓ Los componentes de Java
- ✓ Instalación del JDK y Eclipse IDE
- ✓ Creación y ejecución de un proyecto Java.



Rompehielo 🧊



Compartamos experiencias: 🙌

¿Cómo explicarían qué es un algoritmo en sus propias palabras?



¿Pueden dar ejemplos de algún proceso paso a paso que siguen en su vida cotidiana?

- Les recomendamos abrir el mic 🎤 o escribir por el chat 😊



› Algoritmos: concepto y características



¿Qué es un algoritmo?



**Los algoritmos son una parte fundamental de la programación y la informática.
Pero ¿qué es un algoritmo exactamente?**

Un algoritmo es una secuencia finita de instrucciones bien definidas y no ambiguas, que permite realizar una tarea específica. Es como una receta o manual de pasos para resolver un problema particular. Por ejemplo, las instrucciones para instalar un programa en la computadora, las indicaciones para llegar a un lugar, o los pasos para hacer una tarta, son todos algoritmos.



En informática, los algoritmos nos permiten dar soluciones a problemas de procesamiento de información de manera precisa y automatizada. Son utilizados para el diseño de programas y software.





Partes de un algoritmo

Los algoritmos diseñados **son independientes del lenguaje**, es decir, son simplemente instrucciones que se pueden implementar en cualquier lenguaje y, sin embargo, el resultado será el mismo.

Partes de un algoritmo



- **Entrada:** son todos los datos que se introducen al algoritmo para que pueda ejecutarse en base a dicha información.
- **Procesamiento:** con lo que ha recibido en la entrada, el algoritmo llevará a cabo una serie de cálculos para dar con la solución al problema.
- **Salida:** los resultados que se han obtenido del procesamiento de datos se mostrarán en la salida del algoritmo.



Características de los algoritmos



Principales características:



- **Secuenciales:** se procesan consecutivamente. Responden a una secuencia de pasos determinados.
- **Precisos:** tienen que ser objetivos en la resolución del problema.
- **Ordenados:** deben leerse y ejecutarse en un orden específico.
- **Finitos:** deben contar con un número concreto de pasos.
- **Concretos:** deben mostrar una solución al problema especificado.
- **Definidos:** antes los mismos datos de entrada, siempre debe conseguirse los mismos datos de salida.

› Variables



¿Qué es una variable?



Los datos que vamos a manipular en los algoritmos deben estar contenidos en una variable.

Las variables son **contenedores** que almacenan valores que pueden cambiar. Es un espacio en la memoria de la computadora donde se almacena temporalmente un dato durante la ejecución del programa.



A estas se les asigna un **identificador** o nombre, y el **tipo de dato** que van a almacenar.

El valor de las variables **puede modificarse** durante la ejecución.

Las **constantes** también almacenan valores pero su contenido no cambia durante la ejecución.





Características de las variables



Principales características:

- Deben comenzar con letra.
- No deben contener espacios en blanco.
- No deben contener meta caracteres.
- No se podrán utilizar palabras reservadas como nombres de variable.
- Los identificadores de las variables deben ser únicos e irrepetibles.



Al declarar variables, es recomendable asignar un **valor inicial**:

- Ayuda a evitar errores y resultados inesperados.
- Permite documentar el propósito de la variable según el valor inicial.



Ejemplo: Entero cantidad = 0

Variables locales y globales

Variables Locales:

- Se declaran dentro de un bloque de código como una función o bucle.
- Su ámbito está limitado a ese bloque, no existen fuera de él.
- Se crean y destruyen cada vez que se ejecuta el bloque de código.
- No pueden ser accedidas desde afuera del bloque que las contiene.
- Permiten reutilizar nombres de variables en diferentes partes.

Variables Globales:

- Se declaran fuera de cualquier bloque, al inicio del programa.
- Su ámbito es toda la ejecución del programa, pueden usarse en cualquier parte.
- Se crean al comenzar el programa y se destruyen al finalizar.
- Pueden ser accedidas y modificadas desde cualquier parte del código.
- Deben tener nombres únicos en todo el programa.

› Tipos de Datos

¿Qué son los tipos de datos?

Un dato es cualquier elemento que necesitemos procesar en un programa. Para saber qué operaciones podemos realizar con estos datos, los clasificamos según su tipo.

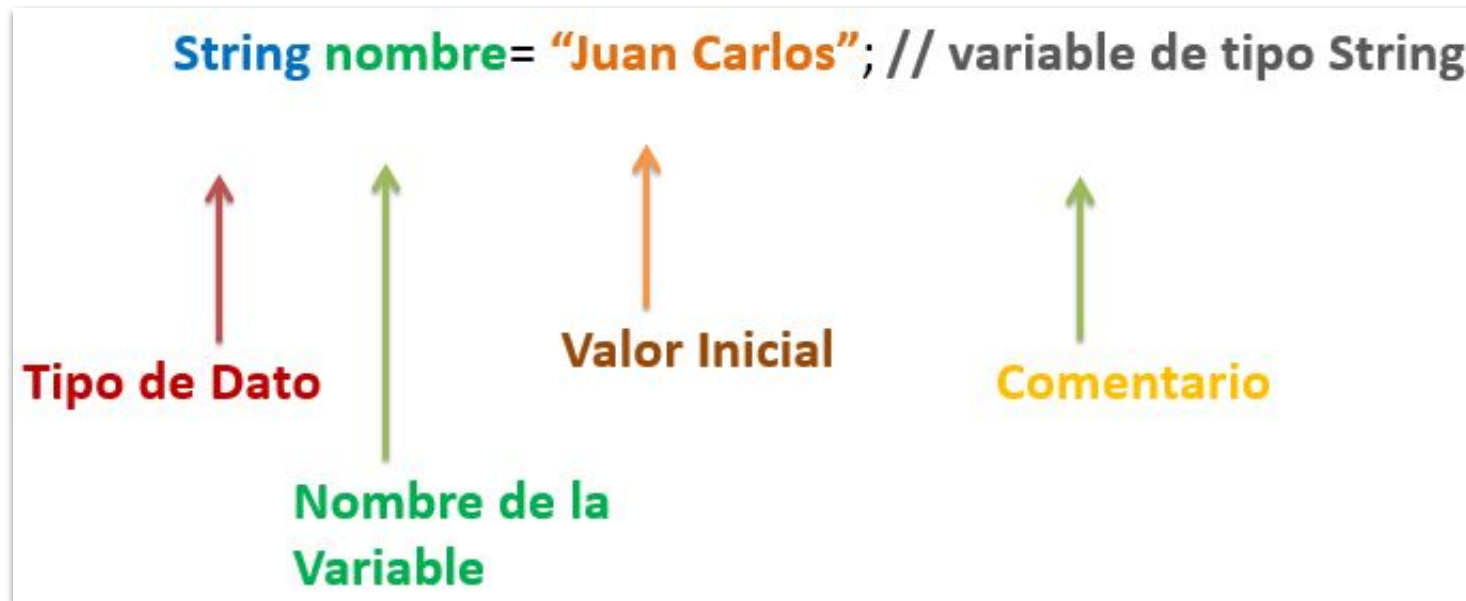
Un tipo de datos es un conjunto de valores que tienen una característica en común y que responden a unas operaciones determinadas.

Dependiendo del lenguaje de programación se puede trabajar con unos tipos u otros. Por ejemplo hay lenguajes que distinguen entre números enteros y números decimales y otros en los que solamente se tiene el tipo de datos numérico, englobando tanto decimales como enteros en el mismo saco.



Tipos de Datos

En Java esta es la forma de declarar una variable con su respectivo tipo de dato



En todos los lenguajes de programación encontramos una clasificación de tipos de datos siempre presente, **los tipos de datos simples y los tipos de datos compuestos**.

Tipos de Datos

SIMPLES:

- ✕ También llamados tipos de datos primitivos, son aquellos que contienen un elemento único de un tipo de datos particular y no se pueden descomponer en varios datos independientes.

Ejemplos:

- **Entero:** 5
- **Real:** 5,5
- **Cadena:** "hola mundo"
- **Caracter:** 'H'
- **Booleano:** Verdadero/Falso

COMPUESTOS:

También llamados tipos de datos complejos o estructurados, se componen de agrupaciones de tipos de datos simples.

Ejemplos:

- **Arreglos:** es un conjunto de valores que se almacenan en una misma referencia de la memoria
- ✕ **Objetos:** podemos tener distintos tipos de datos simples, y además existen operaciones asociadas a esos objetos.



LIVE CODING

Ejemplo en vivo

Escribiendo un algoritmo:

Vamos a escribir un algoritmo que permita calcular el promedio de 3 calificaciones de un alumno.

1. *Definir variables*
2. *Definir tipos de datos*
3. *Leer datos de entrada*
4. *Realizar operaciones necesarias*
5. *Almacenar resultado*
6. *Mostrar resultado por pantalla*

 **Tiempo: 20 minutos**



Ejercicio N° 1

Creando un algoritmo



Nombre del ejercicio N° 1

En salas reducidas, deberán crear un algoritmo que permita resolver una receta de cocina a su elección. Este algoritmo puede escribirse como texto en un archivo doc. Luego, compartir con el resto del curso cuáles fueron las partes más difíciles de definir.

Consigna:

El algoritmo debe tener:

- Datos de entrada: identificar qué variables serán necesarias, y el tipo de dato que guardaran.
- Procesamiento: definir qué hacer con las variables.
- Salida: resultado esperado

¡Atención!

Recuerda que el nombre o identificador debe ser simple y descriptivo.

Tiempo : 25 minutos

○

¿Alguna consulta?

+



RESUMEN

¿Qué logramos en esta clase?

- ✓ Conocer el concepto y las características de los algoritmos.
- ✓ Identificar variables, qué son y cómo declararlas
- ✓ Identificar los tipos de datos, qué son y qué operaciones se pueden realizar con cada uno.



#WorkingTime

Continuemos ejercitando

¡Antes de cerrar la clase! Te invitamos a: 📌 📌 📌

1. Repasar nuevamente la grabación de esta clase
2. Revisar el material compartido en la plataforma de Moodle (lo que se vio en clase y algún ejercicio adicional)
 - a. *Lectura Módulo 4, Lección 2: páginas 1 - 7*
3. Traer al próximo encuentro, todas tus dudas y consultas para verlas antes de iniciar nuevo tema.

¡Muchas Gracias!

Nos vemos en la próxima clase 🙌



Momento: ✚

Time-out!

🕒 5 min.

