

TABLA DE CONTENIDO PARA ENSEÑAR EL LENGUAJE DE PROGRAMACIÓN JULIA

Modulo 1. Primeros pasos: contextualización

- 1 Descripción general
- 2 Características del lenguaje
- 3 Ejemplos de uso
- 4 Otra información de interés

Fuentes:

- [Julia \(programming language\) - Wikipedia](#)
- [Noteworthy Differences from other Languages · The Julia Language](#)

Modulo 2. Conceptos básicos

- 1) Tipos de datos
 - a. Conceptos generales en la programación
 - b. Tipos de datos en Julia
 - i. Enteros y puntos flotantes
 1. Enteros
 2. Punto flotante
 3. Otros conceptos (Aritmética de precisión arbitraria, Coeficientes de literales numéricos)

Imagen de referencia de temas

Integers and Floating-Point Numbers
<ul style="list-style-type: none">◦ Integers◦ Floating-Point Numbers◦ Arbitrary Precision Arithmetic◦ Numeric Literal Coefficients◦ Literal zero and one

Fuente: [Integers and Floating-Point Numbers · The Julia Language](#)

- ii. Números complejos y racionales
 1. Complejos
 2. Racionales

Fuente: [Integers and Floating-Point Numbers · The Julia Language](#)

- iii. Strings
 1. Ideas básicas sobre Strings
 2. Operaciones entre strings
 3. Otros conceptos

Imagen de referencia de temas

Strings
◦ Characters
◦ String Basics
◦ Unicode and UTF-8
◦ Concatenation
◦ Interpolation
◦ Triple-Quoted String Literals
◦ Common Operations
◦ Non-Standard String Literals
◦ Regular Expressions
◦ Byte Array Literals
◦ Version Number Literals
◦ Raw String Literals

Fuente: [Strings · The Julia Language](#)

iv. Tipos

1. Declaración
2. Clasificación y ejemplos

Imagen de referencia de temas

Types
◦ Type Declarations
◦ Abstract Types
◦ Primitive Types
◦ Composite Types
◦ Mutable Composite Types
◦ Declared Types
◦ Type Unions
◦ Parametric Types
◦ UnionAll Types
◦ Singleton types
◦ Types of functions
◦ <code>Type{T}</code> type selectors
◦ Type Aliases
◦ Operations on Types
◦ Custom pretty-printing
◦ "Value types"

Fuente: [Types · The Julia Language](#)

2) Tipos de operadores y operaciones matemáticas

- a. Tipos de operadores
- b. Operaciones matemáticas

Imagen de referencia de temas

Mathematical Operations and Elementary Functions
◦ Arithmetic Operators
◦ Boolean Operators
◦ Bitwise Operators
◦ Updating operators
◦ Vectorized "dot" operators
◦ Numeric Comparisons
◦ Operator Precedence and Associativity
◦ Numerical Conversions

Fuente: [Mathematical Operations and Elementary Functions · The Julia Language](#)

3) Variables

a. Conceptos y estándares de nombramiento

Fuente: [Variables · The Julia Language](#)

b. Clasificación de las variables según su alcance

- i. Global
- ii. Local
- iii. Constantes
- iv. Tipadas

Fuente: [Scope of Variables · The Julia Language](#)

c. Conversiones y promociones

Fuente: [Conversion and Promotion · The Julia Language](#)

4) Estructuras de control

- a. Condicionales
- b. Iteradores
- c. Manejo de excepciones
- d. Expresiones compuestas
- e. Otras

Fuente: [Control Flow · The Julia Language](#)

5) Funciones

- a. Conceptos
- b. Estructura y características

Imagen de referencia

Functions	Methods
<ul style="list-style-type: none">◦ Argument Passing Behavior◦ Argument-type declarations◦ The <code>return</code> Keyword◦ Operators Are Functions◦ Operators With Special Names◦ Anonymous Functions◦ Tuples◦ Named Tuples◦ Destructuring Assignment and Multiple Return Values◦ Property destructuring◦ Argument destructuring◦ Varargs Functions◦ Optional Arguments◦ Keyword Arguments◦ Evaluation Scope of Default Values◦ Do-Block Syntax for Function Arguments	<ul style="list-style-type: none">◦ Defining Methods◦ Method Ambiguities◦ Parametric Methods◦ Redefining Methods◦ Design Patterns with Parametric Methods◦ Parametrically-constrained Varargs methods◦ Note on Optional and keyword Arguments◦ Function-like objects◦ Empty generic functions◦ Method design and the avoidance of ambiguities

Fuente: [Functions · The Julia Language](#) y [Methods · The Julia Language](#)

6) Estructuras de datos

- a. Arreglos y vectores

Fuente que puede dar información útil sobre arreglos: [Multi-dimensional Arrays · The Julia Language](#)

- b. Listas
 - c. Pilas y colas
 - d. Hashes
- 7) Programación Orientada a Objetos (POO)
- a. Conceptos sobre clases y objetos
 - b. Principios
 - c. Estructura y definiciones

8) Interfaces

Fuente: [Interfaces · The Julia Language](#)

Modulo 3. Vista general de otros temas de interés

- 1) Corriendo programas externos
- 2) Guía de estilo para Julia
- 3) Documentación en Julia
- 4) Networking and streams
- 5) Concurrent and parallel programming
 - a. Asynchronous programming
 - b. Multi threading

Modulo 3. Explorando librerías externas

- 1) Biología
- 2) AI
- 3) Mathematics

Fuente: [Julia Packages](#)