



# Glossary

## A

### **Algorithm**

An algorithm is a step-by-step set of instructions or a computational procedure for solving a specific problem or performing a task. Algorithms are fundamental to computer programming.

### **Array**

Data structure that can hold a collection of elements of the same data type. Arrays can be one-dimensional (vectors), two-dimensional (matrices), or multi-dimensional.

## C

### **Coding**

Coding refers to the act of writing and implementing code in a programming language to create software applications, scripts, or functions.

### **Command**

A command is an instruction given to a program or scripting language to perform a specific action or task. Commands are typically entered in a command window or script.

### **Command window**

The command window is an interactive interface in programming environments like MATLAB or Python where you can enter and execute commands in real-time. It's often used for quick tests and exploration.

### **Condition**

Conditions (or conditional statements) are control structures that allow a program to make decisions based on specified conditions. Common examples include if statements and switch statements.



## D

<b>Data type</b>	Data types define the type of data that can be stored in a variable or used in a program. Common data types include integers, floating-point numbers, strings, and booleans.
<b>Deadlock</b>	In programming, a deadlock is a situation where two or more threads or processes are unable to proceed because each is waiting for the other to release a resource. This creates a standstill where none of the involved threads or processes can continue executing. Deadlocks are a common issue in concurrent programming and can be difficult to diagnose and resolve.
<b>Debugging</b>	Debugging is the process of identifying and fixing errors (bugs) in a program to ensure it functions correctly. It involves inspecting code, setting breakpoints, and tracing program execution.
<b>Docstring</b>	A docstring is a string at the beginning of a Python code block that describes its purpose and usage. It serves as documentation for the code, aiding developers in understanding its functionality and usage.

## F

<b>Function</b>	A function is a self-contained block of code that performs a specific task or operation. Functions take input arguments, perform computations, and return results.
<b>Function parameters/arguments</b>	Function parameters (or arguments) are values or variables passed to a function to provide input data for its execution. Functions use these parameters to perform operations.



## G

<b>Global variable scope</b>	Global variable scope refers to variables that are declared and accessible throughout the entire program. These variables can be accessed and modified from any function or block within the program.
<b>GUI (Graphical User Interface)</b>	A GUI is a visual interface that allows users to interact with software using graphical elements such as windows, buttons, and menus, rather than text-based commands. It provides a more user-friendly experience.

## I

<b>IDE (Integrated Development Environment)</b>	Software application that provides comprehensive facilities to computer programmers for software development. It typically includes a source code editor, build automation tools, and a debugger, all integrated into a single user interface. IDEs are designed to streamline the development process by providing features such as syntax highlighting, code completion, and project management tools.
<b>In-built functions</b>	In-built functions (also called built-in or predefined functions) are pre-written functions provided by a programming language or environment to perform specific tasks. Examples include math functions and string manipulation functions.
<b>Indentation</b>	Indentation refers to the practice of adding spaces or tabs at the beginning of lines of code to visually structure and format the code for readability. It is essential for code organization in many programming languages
<b>Input</b>	Input refers to data or information provided to a program during its execution. Input can come from various



sources, such as the user, files, or sensors.

## J

### **JupyterLab**

An interactive development environment for data science, providing a flexible and powerful interface for computational workflows. It supports multiple programming languages through kernels, including Python, R, Julia, Scala, and others, making it versatile for various data analysis and scientific computing tasks.

## L

### **Levels of abstraction (in functions)**

Different layers of complexity in code. High-level abstraction involves using simple, general functions that hide detailed implementation, making the code easier to understand and use. Low-level abstraction involves detailed, specific functions that provide more control over the operations but require a deeper understanding of the code.

### **Library**

A library is a collection of pre-written code, functions, and routines that can be used and reused in programs to perform common tasks, such as mathematical computations, data manipulation, or input/output operations. Libraries help to speed up development and ensure code reliability.

### **Local variable scope**

Local variable scope refers to variables that are declared and accessible only within a specific function or block of code. These variables cannot be accessed outside the function or block where they were defined.

### **Logical indexing**

Logical indexing is a technique used to access and manipulate elements of an array or data structure based



on logical conditions (e.g., selecting elements that meet specific criteria).

### **Loop**

A loop is a control structure that allows a set of instructions to be executed repeatedly based on a specified condition. Common types include for loops and while loops.

## **M**

### **Main script**

A main script is the primary script or program that serves as the entry point for execution in a software application. It typically contains the main logic and control flow of the program, orchestrating the execution of other modules or functions as needed. The main script often initializes the application, handles user input, calls relevant functions or modules, and coordinates the overall execution flow. It acts as the central point of control for the application's behavior.

### **MATLAB**

MATLAB is a high-level programming language and environment commonly used in scientific and engineering fields. It's known for its powerful numerical and mathematical capabilities

### **Modularity**

Modularity in programming is the design principle of breaking down a program into separate, interchangeable components or modules, each responsible for a specific functionality. This enhances code organization, reusability, and maintainability.

## **O**

### **Output**

Output refers to the results or data produced by a program as a response to its execution. Output can be displayed to the user, saved to a file, or used as input for other processes.



## P

### **Plug-in**

Software component that adds specific features or functionalities to an existing program or system. Plug-ins are often designed to extend the capabilities of software applications without modifying the core codebase. They can be installed or removed separately, allowing users to customize their software experience according to their needs.

### **Programming**

Programming is the process of creating and writing code in a programming language to instruct a computer to perform specific tasks or solve problems.

## S

### **Script**

A script is a collection of code written in a scripting language that can be executed sequentially. It usually contains a series of commands and is saved in a file with a specific extension (e.g., .m for MATLAB).

## T

### **Terminal**

A terminal is a text-based interface to interact with a computer's operating system or run command-line programs. It's often used by developers and system administrators for tasks like running commands and scripts.

### **Toolboxes**

Toolboxes are collections of specialized functions, libraries, and tools that extend the capabilities of a programming environment or language for specific domains or tasks, such as signal processing or machine learning.



## V

### **Variable**

A variable is a named storage location used to store and manipulate data in a program. Variables have names, data types, and values.

### **Variable scope**

Variable scope refers to the region of a program where a particular variable is accessible and can be used.

Common scopes include local (inside a function), global (accessible everywhere), and block-level (within a specific code block).

### **Vector**

A vector is a one-dimensional array that can hold a collection of elements, such as numbers or values.

Vectors are often used for data storage and manipulation.

## W

### **Workspace**

The workspace is a storage area in a programming environment where variables and their values are stored and managed during program execution. It includes both global and local scopes.