

# Integrated Development Environments

---

## Introduction

---



### 1. What is an IDE?

A development environment is a set of processes and programming tools that we use to create software. An integrated development environment or **IDE** is an environment in which processes and tools are coordinated, and they provide a common interface for the software development process (or, at least, for writing the code and run it).

The first IDEs were born at the beginning of the 70s, but they became really popular along the 90s. Their first aim is to save time and have more reliability in software processes. They provide the programmer some useful components that increase the efficiency and reduce the coding time.

Some examples of these IDEs are Geany, Visual Studio Code, NetBeans, Eclipse, Visual Studio, IntelliJ... In this unit we are going to focus on some of them.

### 2. IDE types

There are two main types of IDEs:

- **General purpose IDEs:** they can work with many programming languages, and they are (usually) more simple and feature limited. Examples of these IDEs are Geany, Visual Studio Code, Sublime Text...
- **Specific purpose IDEs:** they usually work with one (or a couple of) programming languages, so they are very specific environments, and provide a greater bunch of features for this language. Examples of these IDEs are IntelliJ, Visual Studio, NetBeans, Eclipse...

### 3. Functions and components of an IDE

Development environments are composed by some useful programming tools that help us meet our goals. These tools can be:

- **Source code editor:** this is an essential part of every IDE, since it lets us write the source code of our program. It usually highlights the code syntax, and may also have some additional features, such as code autocompletion, auto indentation, context help, access to API documentation, auto insertion of parentheses, square brackets...
- **Compilers and interpreters:** these tools are in charge of translating our source code into machine code. Some IDEs lets us add many different compilers or interpreters for many different languages, as we will see later.

- **Debugger:** it is in charge of debugging our source code. In other words, it lets us examine step by step the execution of our program, check the values of the different variables and stop the program at any point.
- **GUI builder:** it simplifies the creation of graphical user interfaces (GUI), so that we can place the different controls (buttons, lists, menus...) wherever we want.
- **Version control:** with version control tools we can manage every change in the source code of our projects, so that we will have different versions of each source file, and we can go back and recover any older version at any time. It is an essential tool for teamwork.