When you start programming, an **Integrated Development Environment** (IDE) can help with the main developer workflows allowing you to keep focusing on the language specifics. We encourage you to use **PyCharm**for your Python projects. PyCharm is a dedicated Python IDE that provides a variety of useful features like code completion, code inspections, project navigation, and refactorings.

**Overview**

PyCharm is available in two editions:**Community and Professional**. The Community edition is free and [open-sourced](https://github.com/JetBrains/intellij-community/tree/master/python). It comes with everything you need for smart and intelligent Python development, including code assistance, refactorings, visual debugging, and version control integration. PyCharm Professional is a paid version of PyCharm that is aimed at professional Python, web, and data science development. With PyCharm Professional, you can create web framework applications, develop remotely, analyze big data, and work with Jupyter notebooks. See more details about each edition on the [PyCharm product page](https://www.jetbrains.com/pycharm/" \t "_blank).

**Install and run PyCharm**

Before you start, check the requirements for memory and operating systems in the [PyCharm System Requirements](https://www.jetbrains.com/help/pycharm/installation-guide.html" \l "requirements" \t "_blank).

**Install Python**

PyCharm is a Python-specific IDE, so you need some Python to start your work. Download it from [python.org](https://www.python.org/), then install it according to your operating system requirements.

**Download and install PyCharm**

1. Download PyCharm Community from <https://www.jetbrains.com/pycharm/download>.
2. Run the installer and follow the wizard steps. See more details in the [PyCharm Installation Guide](https://www.jetbrains.com/help/pycharm/installation-guide.html" \t "_blank).

Note that you can opt for a standalone installation or [Toolbox App](https://www.jetbrains.com/help/pycharm/installation-guide.html#toolbox) that is helpful when you need to install several JetBrains IDEs or handle multiple versions of PyCharm.

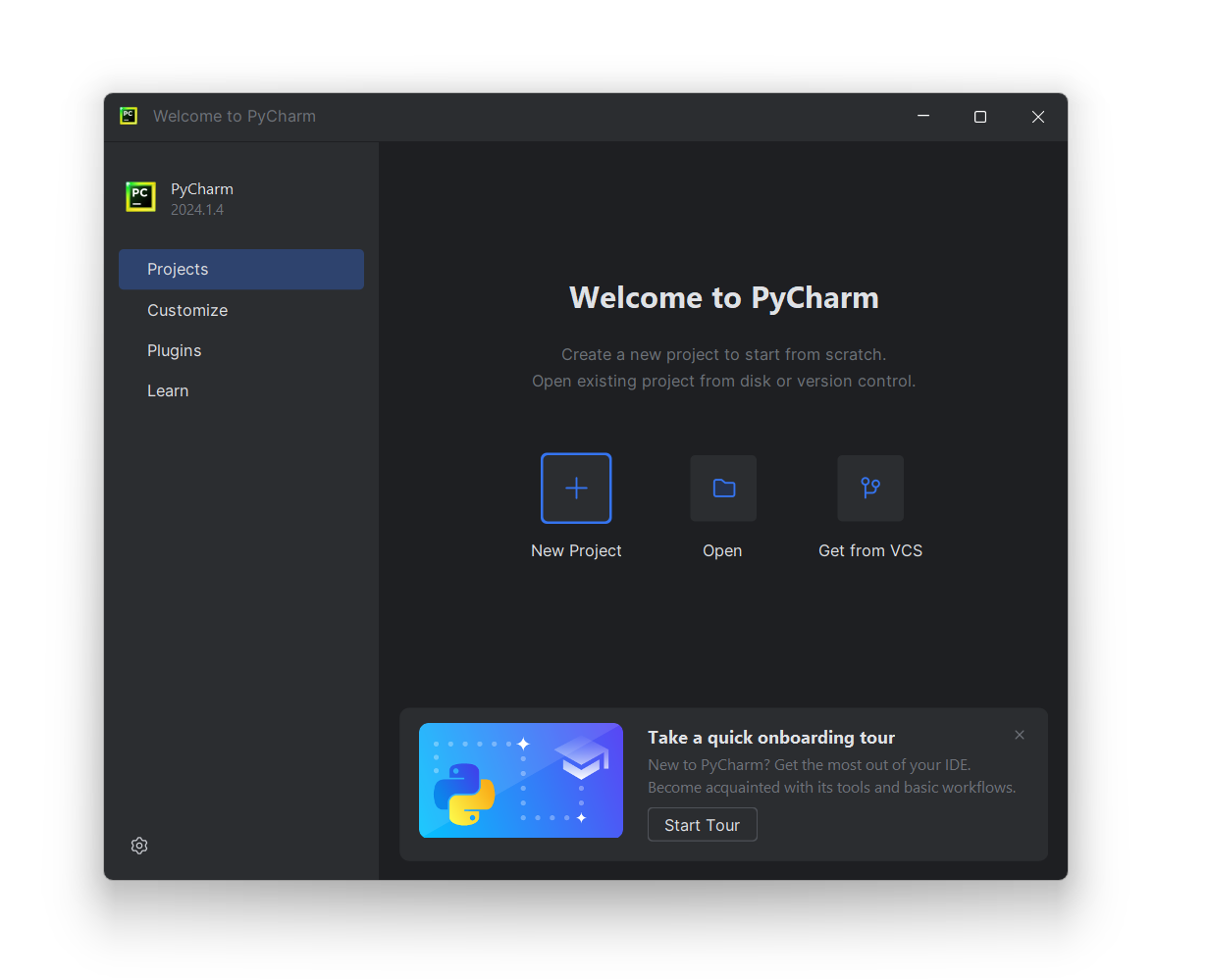
**Launch PyCharm**

To launch PyCharm, perform the following action depending on your operating system:

* **Windows**: Run PyCharm using the Windows Start menu or the desktop shortcut
* **macOS**: Run the PyCharm app from the *Applications* directory
* **Linux**: Run the pycharm.sh shell script in the installation directory under bin

**Welcome screen**

Once you launch PyCharm, you will see the Welcomescreen:



On the Welcomescreen, you can find all options and settings to start your work in PyCharm. The quickest way is to create a new project from scratch or open an existing project from a local drive or a repository.

The start page also contains a link to the **PyCharm onboarding tour** that will help you get acquainted with the main developer's workflow in just 7 minutes. Take the tour to make your start even smoother.

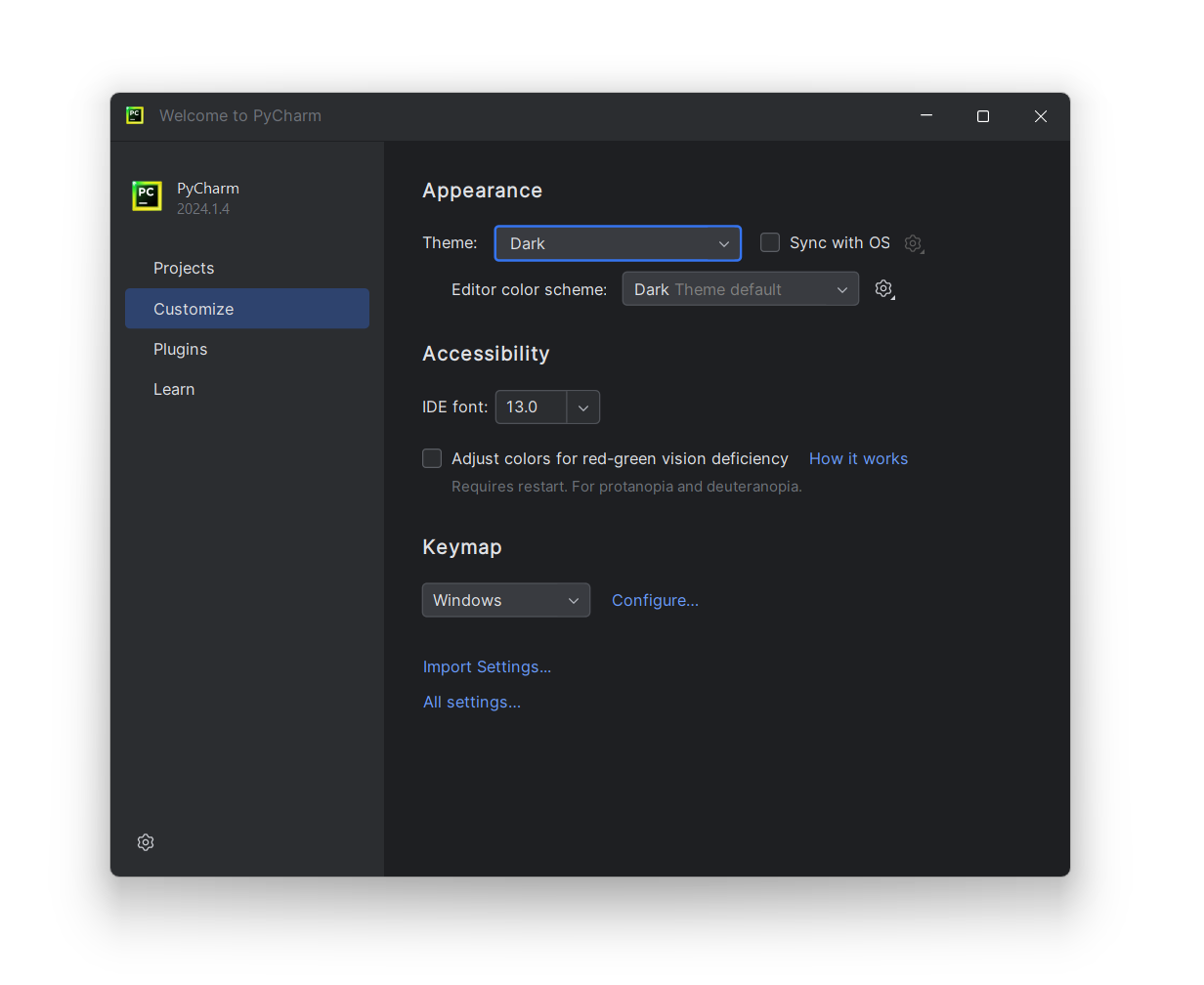
**Quick start**

You can begin your work in the IDE without setting anything beforehand because PyCharm provides almost all features out-of-the-box. Nevertheless, whatever you do in PyCharm, you do in the context of a project. A project is an organizational unit that serves as a basis for coding assistance, bulk refactoring, and coding style consistency. So, you need to start with a project setup. You have the following options:

* [Create a new project](https://www.jetbrains.com/help/pycharm/creating-and-running-your-first-python-project.html) — create a new project in your file system and configure a Python environment using your Python installation.
* [Open an existing project](https://www.jetbrains.com/help/pycharm/open-projects.html#open-local-project) — open an existing PyCharm project. You can also open any folder in your file system, and PyCharm will detect any previously configured Python environment.
* [Get a project from a version control system](https://www.jetbrains.com/help/pycharm/open-projects.html#git) — clone a project from a Git, Mercurial, or Subversion repository.

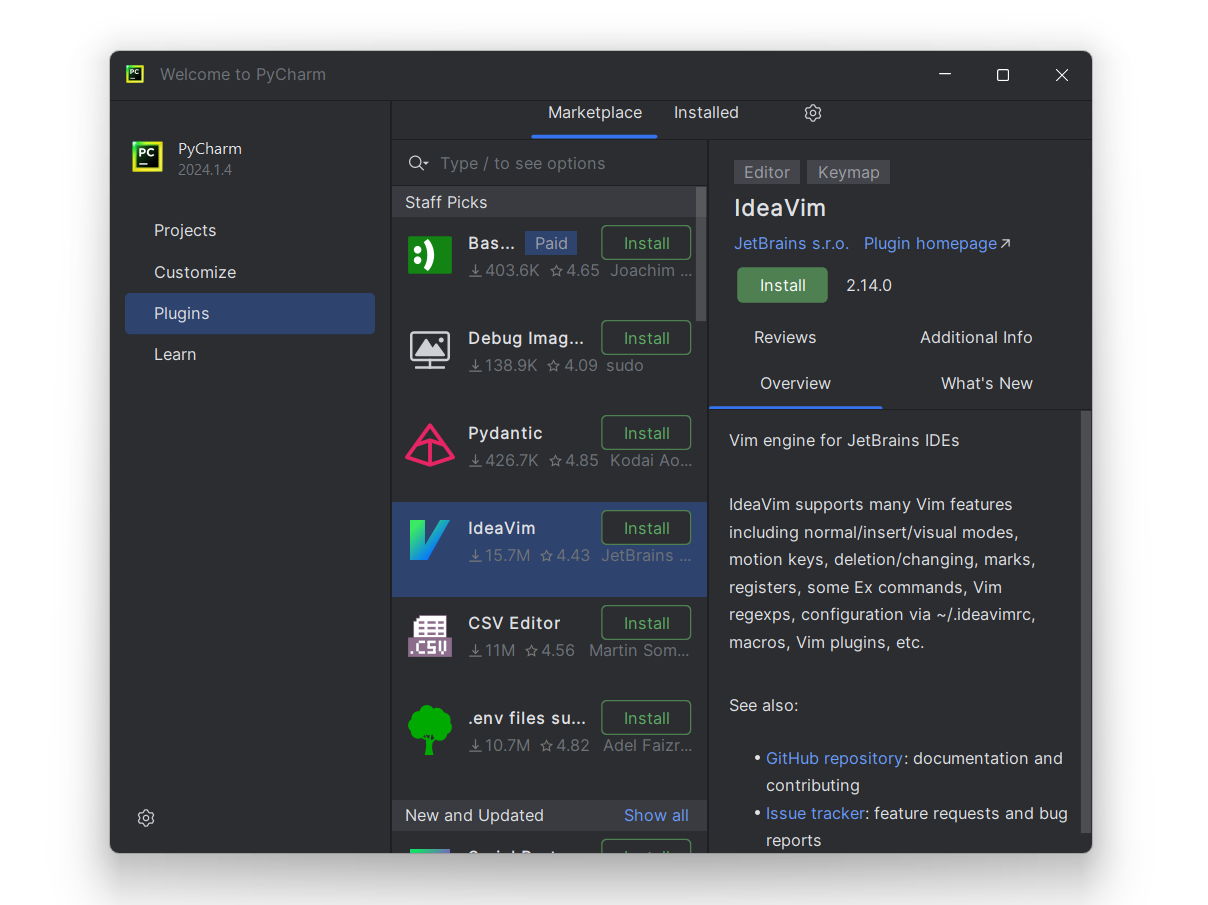
**Customized start**

On the Welcomescreen, you can adjust the IDE default settings. Click **Customize**and select another color theme or select the **Sync with OS** checkbox to use your system's default theme. Here you can also configure accessibility settings or select another keymap.



You can also click **All settings** to open the settings dialog. The settings that you modify at this moment will become the new default configuration for your projects and the IDE.

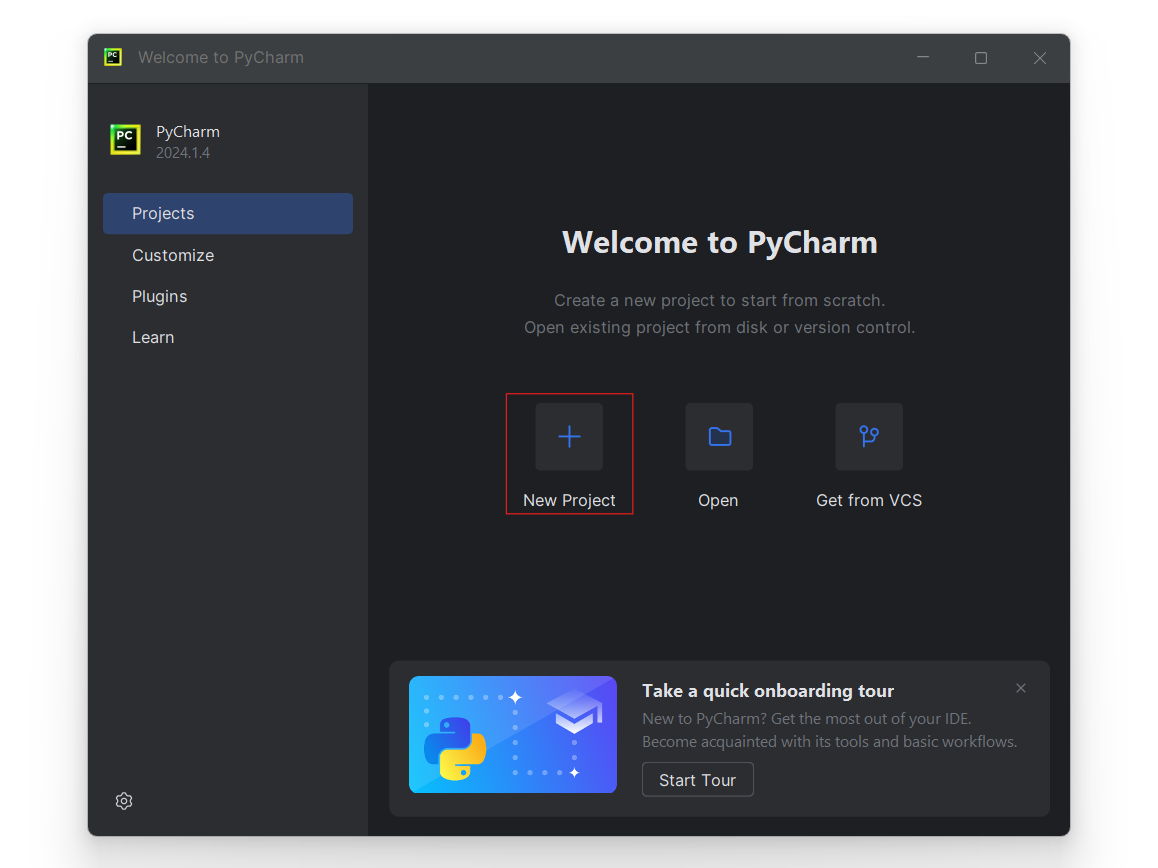
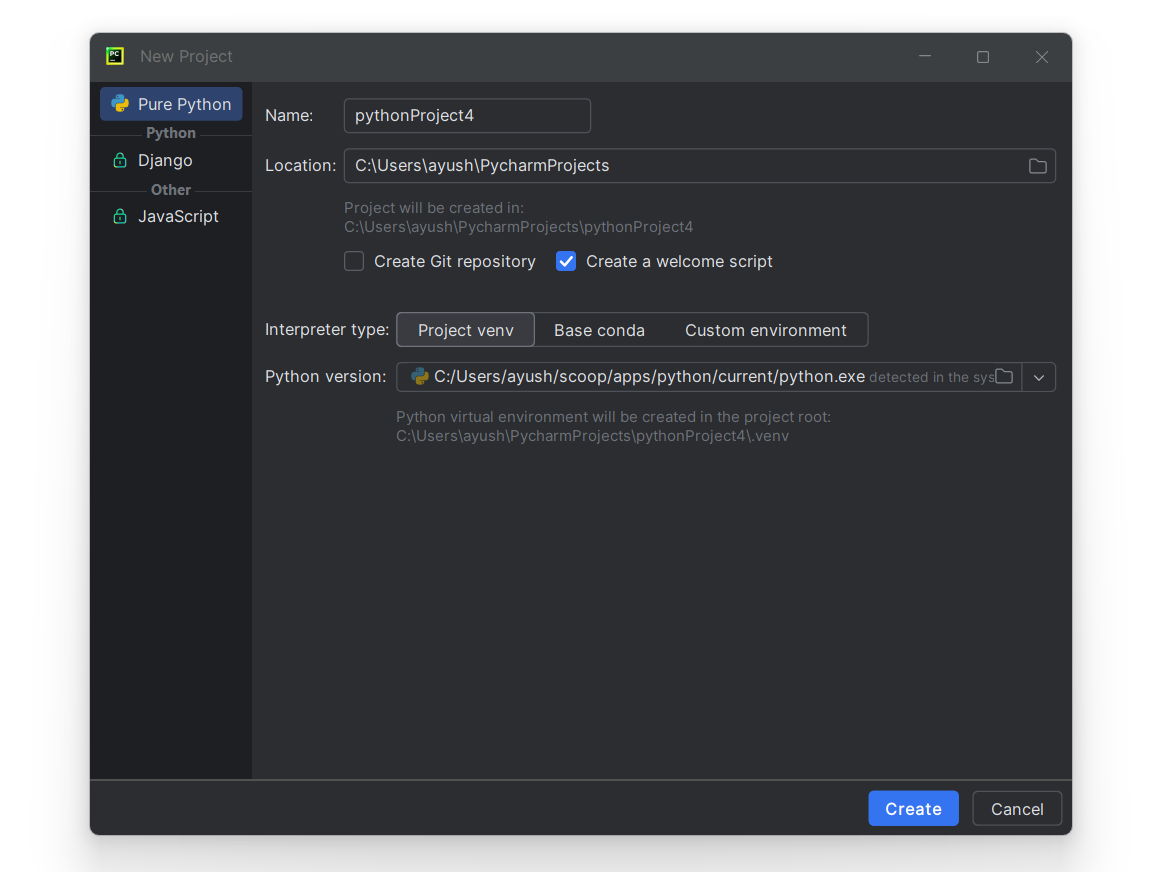
Although PyCharm Community comes with the all main features, you can extend them by installing plugins. Click **Plugins** in the left-hand pane of the Welcomescreen, then download and install the required plugins from the [PyCharm plugins repository](https://plugins.jetbrains.com/pycharm" \t "_blank).



With this, you're ready to create your first Python application.

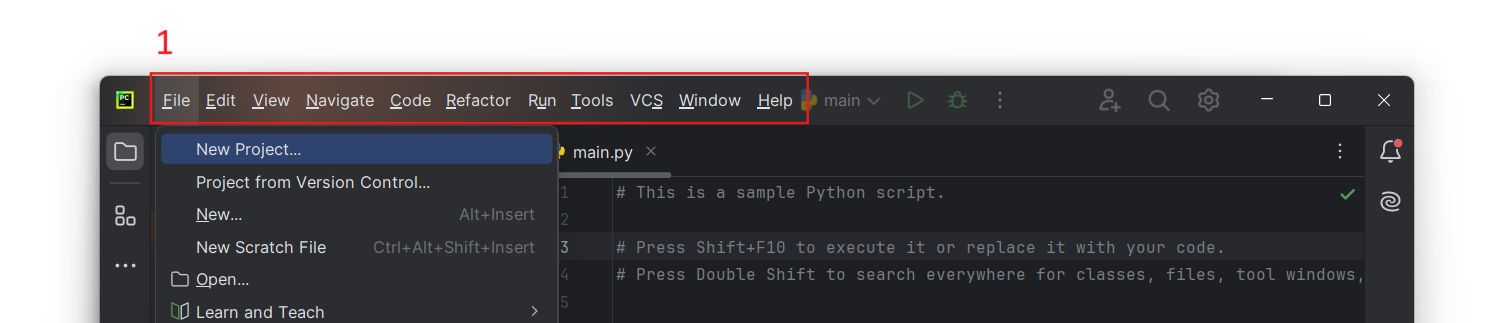
**Create a Python project in PyCharm**

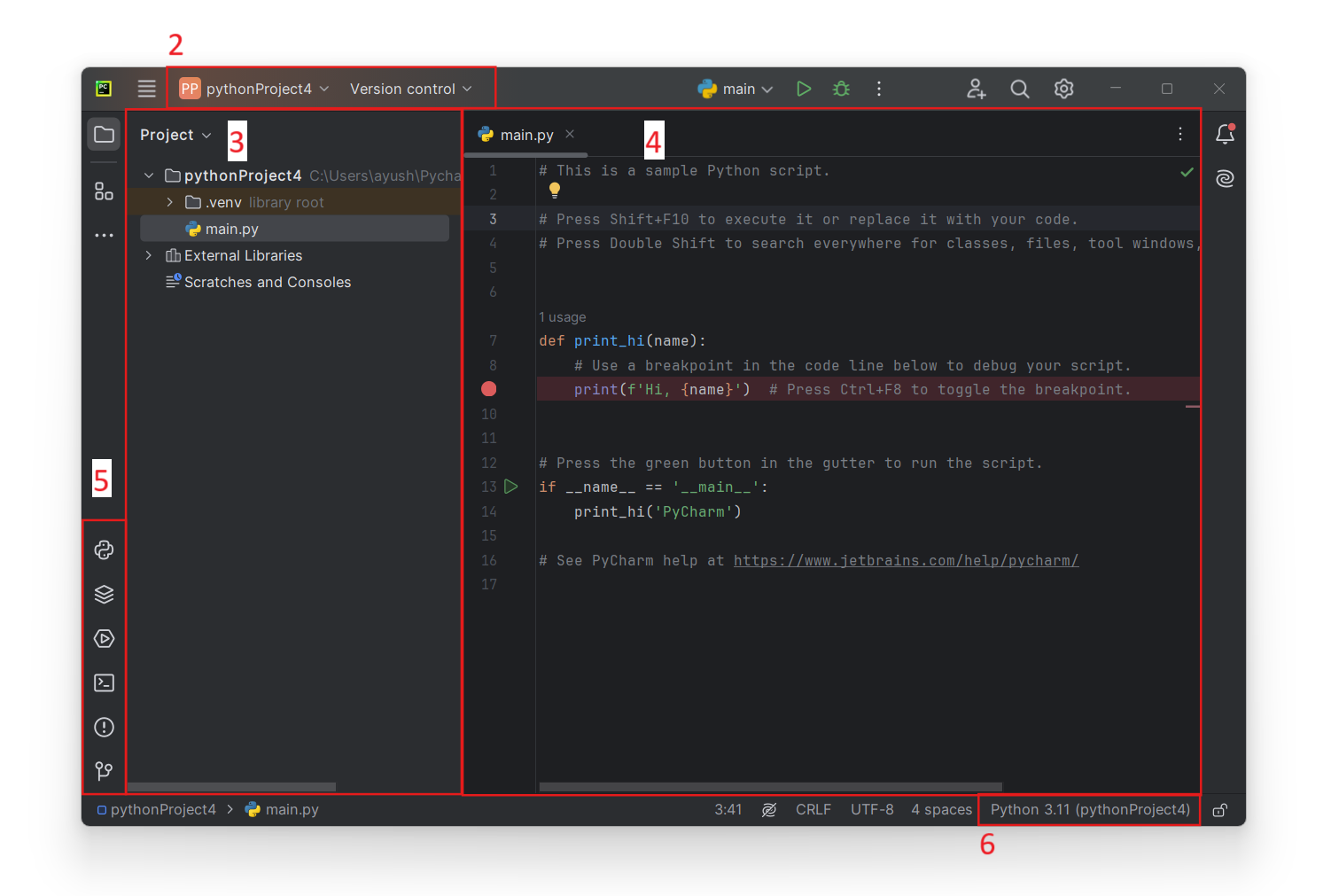
To create a new Python project, perform the following steps:

1. On the Welcome screen, select New Project:
2. In the New Project dialog, PyCharm fills all the required fields for you. The IDE will create a Python virtual environmentandthe Python built-in package manager, which allows you to install external libraries. The Base interpreter list indicatesthe path to the actual Python, which you download and install on your computer.
3. So, at this point, you can select one type of environment: Virtual Environment (default), Conda, Pipenv, or Poetry, and create it using one of the Python interpreters installed in your system. Refer to [Creating Python Projects](https://www.jetbrains.com/help/pycharm/creating-empty-project.html) for more details about other project options.
4. Mind the **Create a main.py welcome script** checkbox. It is selected by default. With this option enabled, PyCharm creates a Python file with some basic code that can be a good starting point for your application.
5. Click **Create**to complete the task.

When the project is created, the *main.py* file is opened in the editor. This file shows the basic script and provides useful hints so that you can start editing code in PyCharm. For the time being, remember one of the most [helpful shortcuts](https://www.jetbrains.com/help/pycharm/searching-everywhere.html) Shift + Shift. It opens a dialog where you can find any command, setting, code construct, or project file. Refer to [Working with Source Code in PyCharm](https://www.jetbrains.com/help/pycharm/working-with-source-code.html) for more detail on available coding assistance.

You can add more files of various types to your project, as well as directories and Python packages. They are all listed in the Project tool window. Refer to the following PyCharm web help topic for more details about adding files and directories to a project: [Populating PyCharm Projects](https://www.jetbrains.com/help/pycharm/populating-projects.html).

You already learned about the PyCharm editor and Project tool window. Let's look at the entire PyCharm user interface:



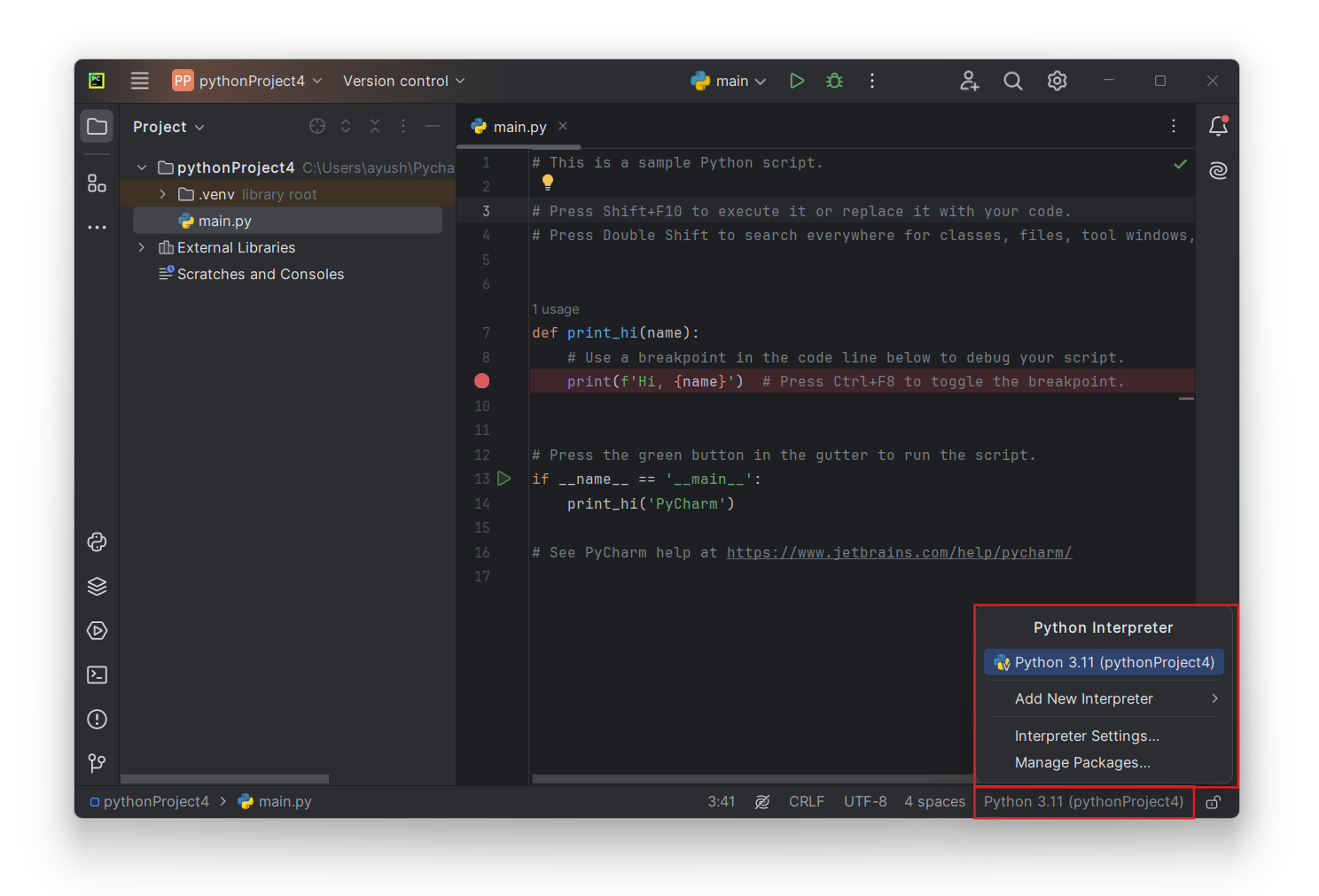
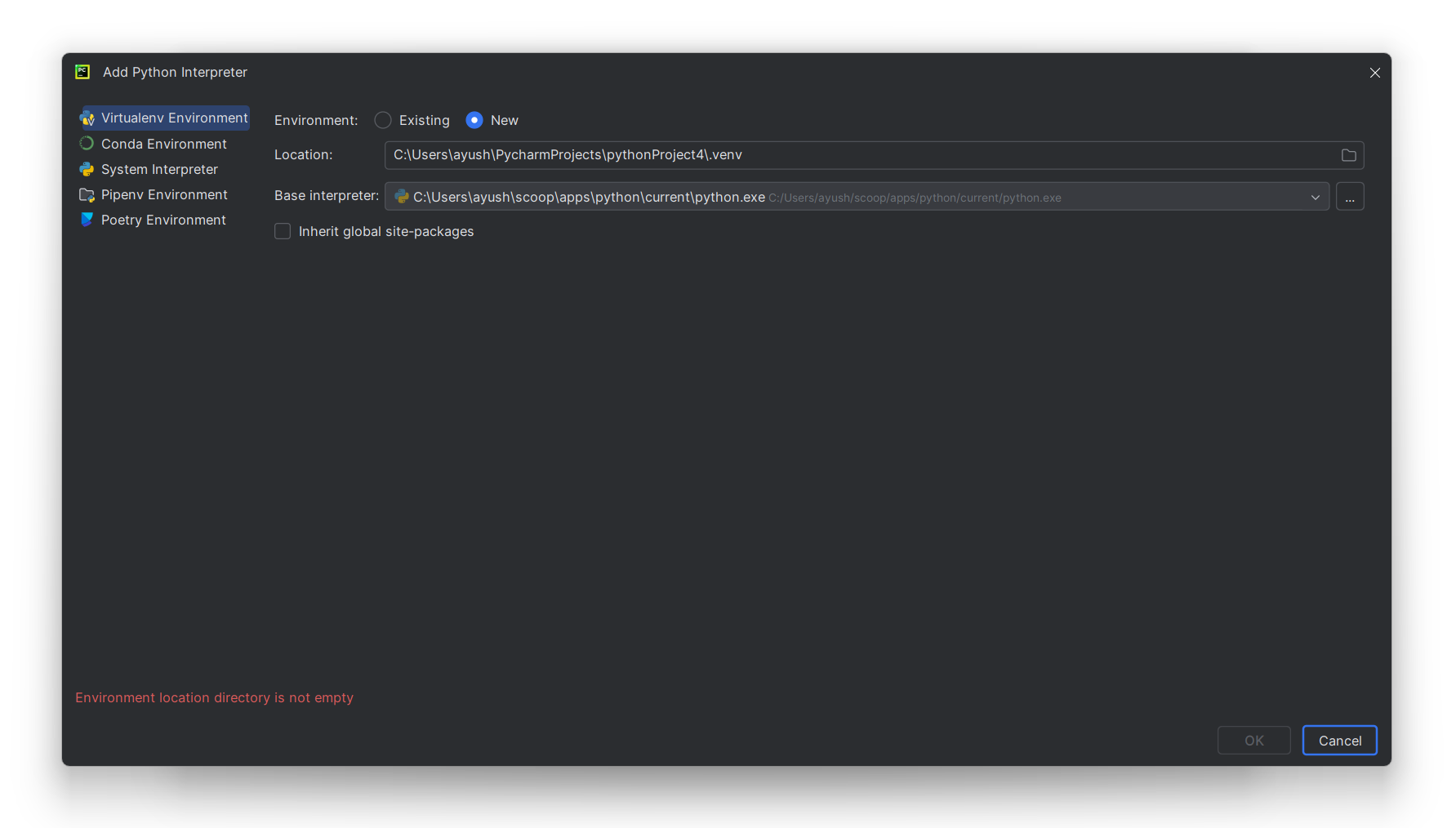
The main elements of the IDE UI are:

1. Main menu. Its position is OS-specific;
2. The navigation bar and main toolbar;
3. Project tool window;
4. Editor;
5. Tool windows that include the Python Console and Python Packages windows;
6. Python interpreter selector. It shows the environment that is currently configured for the project.

After doing this, you have created a PyCharm project and configured a Python interpreter for it. You can create more Python interpreters if you need various environments to run your scripts (for example, when you want to execute them on different Python versions).

**Create Python interpreters**

The easiest way to open the Python interpreter settings is to click the **Python Interpreter** selector located in the lowest part of the PyCharm window, on its Status Bar.

1. Click the interpreter selector. You should be able to see all the interpreters that have been configured to be used by all projects. 
2. Select **Add New Interpreter > Add Local Interpreter**from the menu.
3. You can see various types of interpreters in the **Add Interpreter** dialog:
4. You can create a new interpreter or continue to use an existing one.

For a new interpreter, select **Base interpreter** from the list, or click "..." and find a Python executable in your file system. For an existing interpreter, select **Interpreter**, or, similarly, click "..." to discover a Python executable.

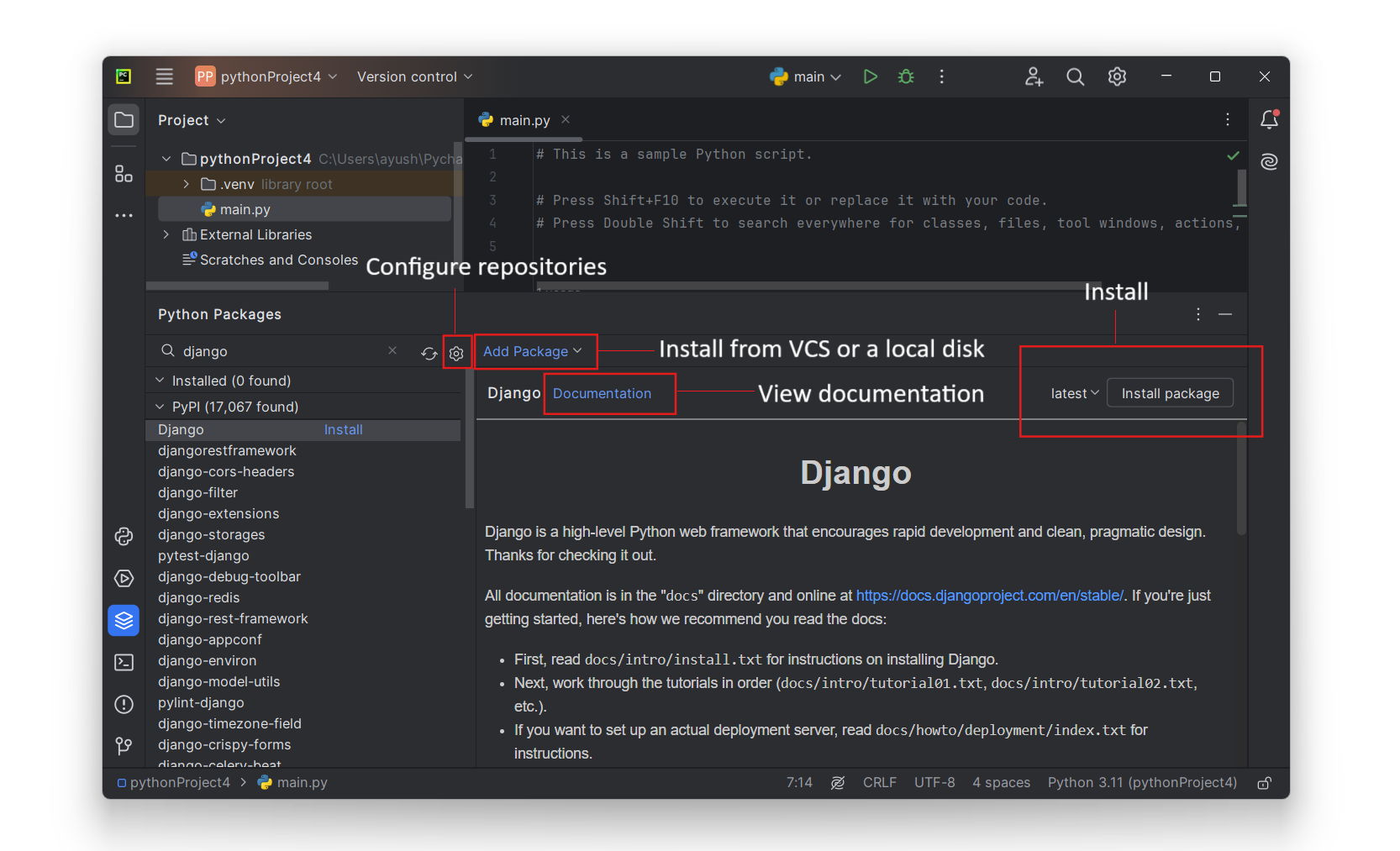
If PyCharm is not able to detect Python on your machine, it provides two options: download the latest Python versions from [python.org](https://www.python.org/) or specify a path to the Python executable (in case of non-standard installation). See more information and related procedures in [Configure a Python Interpreter in PyCharm](https://www.jetbrains.com/help/pycharm/configuring-python-interpreter.html).

When you save the changes, the newly created interpreter is set for the current project.

The key point for creating different isolated environments is to keep different sets of Python packages. So, let's learn how to install packages in PyCharm.

**Install Python packages**

The Python Packages tool window provides the quickest and neat way to preview and install packages for the currently selected Python interpreter.



To install a package from the default repository:

1. Type the package name in the **Search**field of the Python Packages tool window.
2. Locate the package in the list of the default repository
3. Click the **Install**button in the upper-right corner of the tool window

For more details about installing Python packages, see [Install, Uninstall, and Upgrade Packages in PyCharm](https://www.jetbrains.com/help/pycharm/installing-uninstalling-and-upgrading-packages.html).

Read more on this topic in [Testing Python Code 101 with PyTest and PyCharm](https://hyperskill.org/blog/post/testing-python-code-101-with-pytest-and-pycharm#further-reading-and-resources) on Hyperskill Blog.

**Conclusion**

To sum up:

* PyCharm is a Python-specific IDE. It has two editions: Community (free) and Professional (30-day free trial).
* To install PyCharm, download an installer from <https://www.jetbrains.com/pycharm/download>
* The Welcome screen provides a quick way to create, open, or clone a Python project.
* You can customize your IDE with PyCharm settings.
* To start your way in PyCharm, you need to create a project, configure a Python virtual environment (Python interpreter), and add your Python code to the main.py file or to other Python files.
* PyCharm allows you to configure various Python interpreters depending on your operating system and edition of PyCharm.
* With the Python Packages tools window, you can quickly install packages on the selected Python interpreter.
* Find more details and specific procedures in the [PyCharm Web Help](https://www.jetbrains.com/help/pycharm/" \t "_blank).

What shortcut will prompt a dialog where you can find any command, setting, code construct, or a project file?

Shift + Shift (pressing shift twice)

PyCharm Community

Free and open-sourced; comes with everything you need for smart and intelligent Python development, including code assistance, refactorings, visual debugging, and version control integration.

PyCharm Professional

Paid version of PyCharm that is aimed at professional Python, web, and data science development. With it you can create web framework applications, develop remotely, analyze big data, and work with Jupyter notebooks.

Lunch method:

 **Windows** → ✅ *Start menu/desktop shortcut*

 **macOS** → ✅ *Run the app from Applications directory*

 **Linux** → ✅ *Run pycharm.sh script in installation directory*

How can you add more files, directories, and Python packages to your project?

By using the Project tool window

A dependency is A library available and needed for a specific project

What is the main organizational unit called in PyCharm? Project

Hint a collection of classes is called PROJECT!!!