# Hello World!

In this example, we'll get a little familiar with using Python in the terminal.

## Why?

It's important to understand how to get your code running so that you can run your code on any computer. This is especially important when you're working on a team and you need to share your code with others.

## Steps---
### 1. Open your terminal and navigate to the folder where you'll be keeping your notes.
### 2. type **python** for windows and for mac **python3** in the terminal and you should see something like this:

```
Python 3.11.3 (tags/v3.11.3:f3909b8, Apr  4 2023, 23:49:59) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

### 3. Type **print("Hello World!")** and you should see something like this:

```
Python 3.11.3 (tags/v3.11.3:f3909b8, Apr  4 2023, 23:49:59) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello World!")
Hello World!
>>>
```

# Congratulations! You've just run your first Python program!

This was done in the "Python Interpreter" which is a program that allows you to run Python code. This is a great way to test out small snippets of code, but it's not a great way to write a program.

### 4. Exit the Python Interpreter by typing **`exit()`** and pressing enter.

### 5. Create a new file called **hello_world.py** and open it in your text editor, we'll be using **[Visual Studio Code](https://code.visualstudio.com/)** in this course.

## How to access files using terminal

| Step | macOS (Terminal) | Windows (CMD / PowerShell) |
|---|---|---|
| **1. Open terminal** | Open **Terminal** (Applications → Utilities → Terminal) | Open **Command Prompt** (type `cmd` in Start) or **PowerShell** |
| **2. Check where you are** | `pwd` (shows `/Users/yourname`) | `cd` (CMD) or `Get-Location` (PowerShell) (shows `C:\Users\YourName`) |
| **3. Go to Desktop** | `cd Desktop` | `cd Desktop` |
| **4. See what's there** | `ls` | `dir` |
| **5. Enter your project folder** (e.g., `PythonClass`) | `cd PythonClass` | `cd PythonClass` |
| **6. Check for the file** | `ls` (should show `hello_world.py`) | `dir` (should show `hello_world.py`) |
| **7. Run the file** | `python3 hello_world.py` | `python hello_world.py` |

## Basic Terminal Commands: Mac vs Windows

| Action | macOS / Linux (Terminal) | Windows (CMD / PowerShell) |
|---|---|---|
| Show current folder | `pwd` | `cd` (CMD) or `Get-Location` (PowerShell) |
| Go to Desktop | `cd Desktop` | `cd Desktop` |
| List files/folders | `ls` | `dir` |
| Go into your project folder | `cd PythonClass` | `cd PythonClass` |
| Check file exists | `ls` | `dir` |
| Run Python file | `python3 hello_world.py` | `python hello_world.py` |

--- in your terminal, double-check that **hello_world.py** exists by typing **`ls`** if you're on Mac and **`dir`** if you're on Windows and pressing enter.

You should see something like this (on Windows):

```
C:\path\to\your\project> dir
 Volume in drive C is Windows
 Volume Serial Number is 4445-6B5C

 Directory of C:\path\to\your\project

05/26/2023  10:54 AM    <DIR>          .
05/26/2023  10:54 AM    <DIR>          ..
05/26/2023  10:54 AM                 0 hello_world.py
05/26/2023  10:53 AM             2,241 README.md
               2 File(s)          2,241 bytes
               2 Dir(s)  243,710,267,392 bytes free
```

Note: On Mac it looks pretty similar, the most important thing is that you have a file called **hello_world.py** in your project folder.

--- To create an empty file for mac **touch file_name** and for windows **type nul > file_name**

### 6. Type **print("Hello World!")** into your **hello_world.py** file and save it.

--- You can also add a comment to your file by typing `#` and then your comment. Comments are ignored by the Python Interpreter and are used to explain what your code is doing.
--- Your file **hello world.py** should look something like this:

```
# My First Python Program
print("Hello World!")
```

### 7. Run your project by typing **python hello_world.py** for windows and for Mac python3 hello_world.py and pressing enter. You should see something like this:

```
C:\path\to\your\project> python hello_world.py
Hello World!
```

# Congratulations! You've just run your first Python program from a file!

### 8. Let's take a look at what an error looks like because you'll be seeing a lot of them. Change your **hello_world.py** file to look like this:

- Let's make a mistake and see what happens.

```
# My First Python Program
print("Hello World!"   # removed the closing parenthesis
```

- As you can see here I removed your closing parenthesis `)` from the `print` function. This is a common mistake that you'll make when you're first starting out.
- Let's see what happens when we try to run this program.

```
C:\path\to\your\project> python hello_world.py
```

The output you'll see looks something like this

```
  File " C:\path\to\your\project\hello_world.py", line 2
    print("Hello World!"
          ^
SyntaxError: '(' was never closed
```

- This is the "traceback" and is telling us we have an error called a `SyntaxError` and it's telling you that you have a problem with your syntax. In this case, it's telling you that you forgot to close your parenthesis `)`. We'll learn more about this later.

- Something really important to note here is that it tells us the line to look at. This is really helpful when you're trying to figure out what's wrong with your code. So here it tells us our error is on line 2 of the file **hello_world.py**.

### 9. Let's fix our program and go on to the next example!
- Let's add the closing parenthesis `)` back to our `print` function (on line 2) in our hello_world.py file.

```
# My First Python Program
print("Hello World!")
```

## Conclusion

We'll be doing this essentially every day in this course. You'll be writing code, running it, and fixing errors. This is a very important skill to learn and will help you become a better programmer.

# Exercises Intro to Programming

**1. Which of these will print Hello, World! in Python?**

a) print(Hello, World!)
b) echo("Hello, World!")
c) print("Hello, World!")
d) printf("Hello, World!")

**2. Which one is correct?**

a. printt("hi there")
b. print("my name is "
c. print("The computer running this code")
d. print("hi there")

**3. Create a new file** called `new_file.py`.

- In that file, write a Python program that prints your name.
- Save the file and run it from the terminal.
- Make sure the program prints your name correctly.

4. **Create a Python file** called `math_task.py`.

- Write a program that prints the result of adding `25 + 17`.
- Run the program in the terminal and check the output.