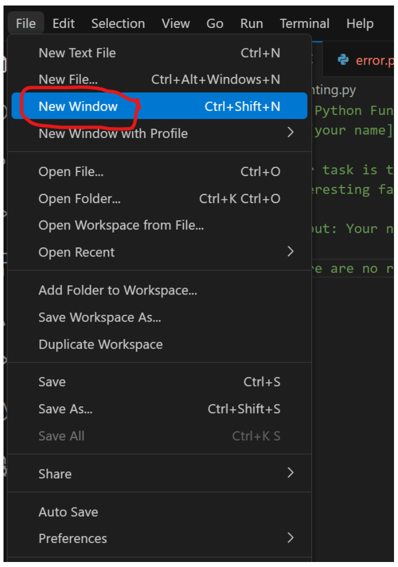
Python Fundamentals 1 - Homework Folder Brief

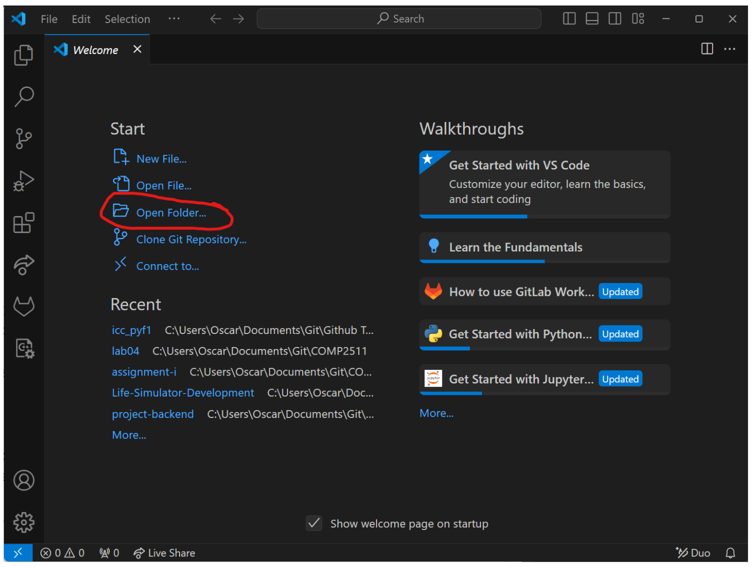
Greetings, young programmers, and Welcome to ICC Python Fundamentals! This is your homework folder, and in here, you will find instructions for all of your homework, as well as tests and some example programs. There will also be worksheets that you can print out and do, in case you missed out on it in class!

How to get started with homework

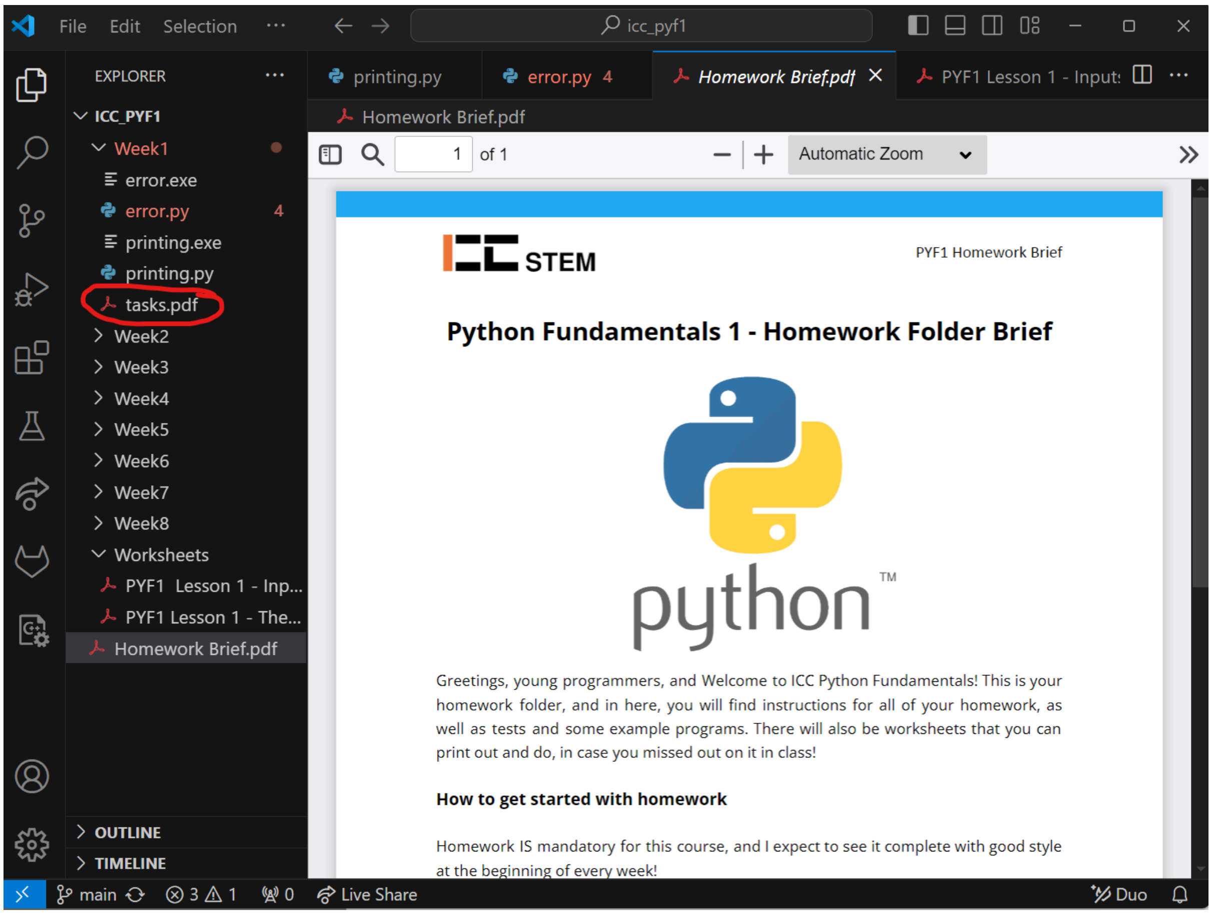
Homework IS mandatory for this course, and it is expected to be completed every week fully working with good style!

To get started with your homework, open up this folder on Visual Studio, NOT on your computer! To do this, simply open up Visual Studio, and the window below shall appear. If not, click on File on the bar at the top, and then New Window.




File Edit Selection
Search
Welcome
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
Recent
icc_pyf1 C:\Users\Oscar\Documents\Git\Github T...
lab04 C:\Users\Oscar\Documents\Git\COMP2511
assignment-i C:\Users\Oscar\Documents\Git\CO...
Life-Simulator-Development C:\Users\Oscar\Doc...
project-backend C:\Users\Oscar\Documents\Git\...
More...
Walkthroughs
Get Started with VS Code
Customize your editor, learn the basics, and start coding
Learn the Fundamentals
How to use GitLab Work... Updated
Get Started with Python... Updated
Get Started with Jupyter... Updated
More...
Show welcome page on startup
Live Share
Duo

Click on "Open Folder" (circled above), then locate your homework folder. Once you have done that, you should now see this window.



PYF1 Homework Brief

Python Fundamentals 1 - Homework Folder Brief

Greetings, young programmers, and Welcome to ICC Python Fundamentals! This is your homework folder, and in here, you will find instructions for all of your homework, as well as tests and some example programs. There will also be worksheets that you can print out and do, in case you missed out on it in class!

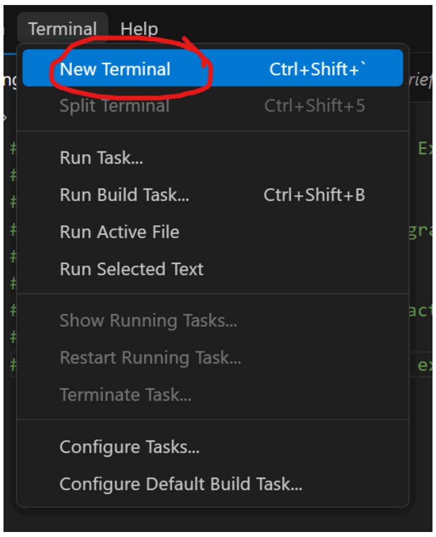
How to get started with homework

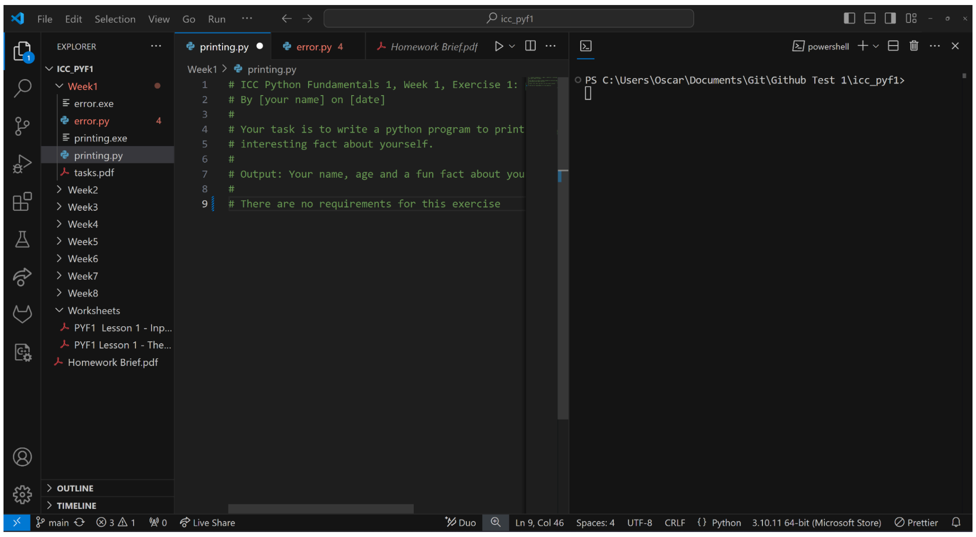
Homework IS mandatory for this course, and I expect to see it complete with good style at the beginning of every week!

Your task now should be to read the tasks.pdf file of the week that you are currently in, and then complete the .py files assigned to you.

Tools to help with homework

Don't just guess whether or not your code is correct or not! To test if your code works and matches up with the specifications, make sure to do a test run of it first! This is done by opening up the terminal, by clicking "Terminal" at the bar on the top and then "New Terminal", or by pressing Ctrl+Shift+` (` is the key on the top left hand side of your keyboard, just below the escape key). Your screen should now look a little something like this.

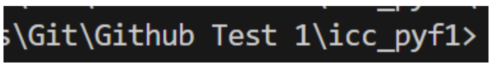




```
# ICC Python Fundamentals 1, Week 1, Exercise 1:
# By [your name] on [date]
#
# Your task is to write a python program to print
# interesting fact about yourself.
#
# Output: Your name, age and a fun fact about you
#
# There are no requirements for this exercise
```

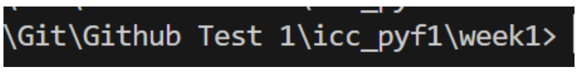
```
PS C:\Users\Oscar\Documents\Git\Github Test 1\icc_pyf1>
```

To test your program in the terminal, please make sure your terminal is in the right week! You can tell which week that the terminal is currently in by looking at the current path. For example, if you want to go to week 1,



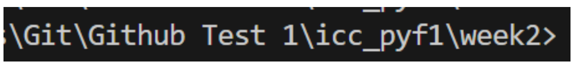
```
s\Git\Github Test 1\icc_pyf1>
```

Your terminal is currently not in any week. Type in cd Week1.



```
\Git\Github Test 1\icc_pyf1\week1>
```

Your terminal is in the correct week! You may now move on to the next step.

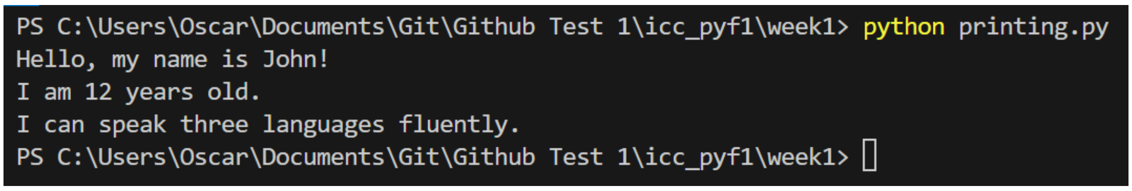


```
\Git\Github Test 1\icc_pyf1\week2>
```

Your terminal is in the wrong week! Type in cd ../Week2.

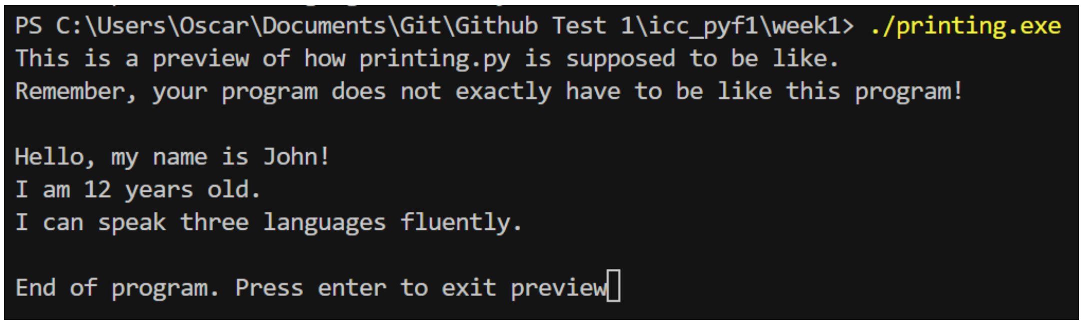
After your terminal has entered the correct week, you can now test your program by typing in python, followed by the name of your python program (e.g. printing.py). For example, to run printing.py, type in python printing.py.

If done correctly, your terminal will now say something like this!



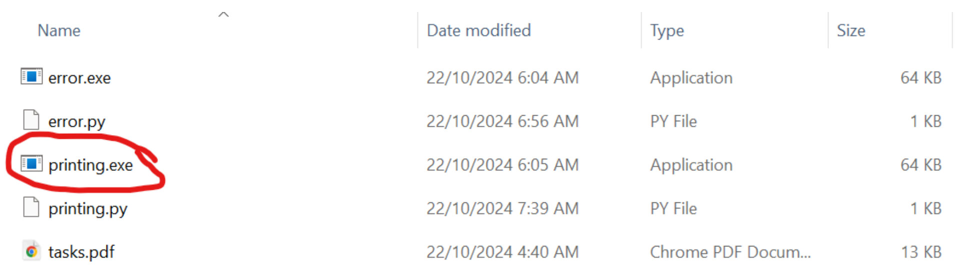
```
PS C:\Users\Oscar\Documents\Git\Github Test 1\icc_pyf1\week1> python printing.py
Hello, my name is John!
I am 12 years old.
I can speak three languages fluently.
PS C:\Users\Oscar\Documents\Git\Github Test 1\icc_pyf1\week1>
```

To test if your output is as it should be, you can also run the corresponding .exe file to check. In this case, printing.exe is our checking file, and to run it, simply type in ./printing.exe. The following should now appear:

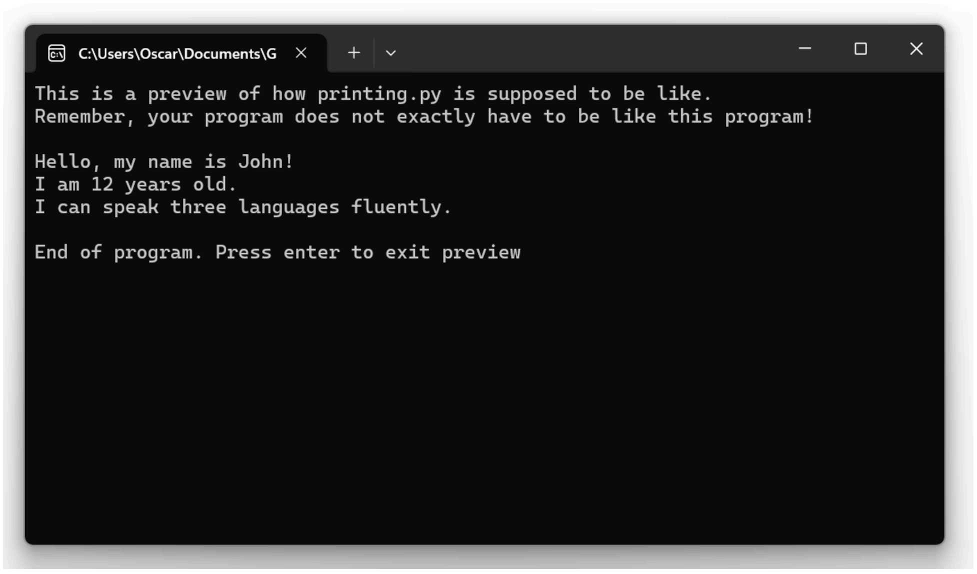


```
PS C:\Users\Oscar\Documents\Git\Github Test 1\icc_pyf1\week1> ./printing.exe
This is a preview of how printing.py is supposed to be like.
Remember, your program does not exactly have to be like this program!

Hello, my name is John!
I am 12 years old.
I can speak three languages fluently.

End of program. Press enter to exit preview
```

Another way to test it is to simply click on it in your folder. A new terminal should then appear.



Name	Date modified	Type	Size
error.exe	22/10/2024 6:04 AM	Application	64 KB
error.py	22/10/2024 6:56 AM	PY File	1 KB
printing.exe	22/10/2024 6:05 AM	Application	64 KB
printing.py	22/10/2024 7:39 AM	PY File	1 KB
tasks.pdf	22/10/2024 4:40 AM	Chrome PDF Docum...	13 KB



```
This is a preview of how printing.py is supposed to be like.
Remember, your program does not exactly have to be like this program!

Hello, my name is John!
I am 12 years old.
I can speak three languages fluently.

End of program. Press enter to exit preview
```

Before you finish...

Make sure that your program obeys all of the styling rules that we have gone over in class! These include:

1. There must be spaces where necessary
2. There must be comments where necessary (Complicated code, functions etc.)
3. Different categories of code must be separated with an extra line (paragraphed)
4. File, variable and function names must be descriptive
5. Nested code must be indented
6. Outputs must have good punctuation
7. Each line of code should be under 80 characters, MUST be under 100 characters

After completing your homework

Congratulations! Now you just wait until the beginning of the next lesson for it to be checked. Remember to read the requirements carefully, otherwise you might lose marks!