

Python Coding



<https://pellcorp.github.io/coding/docs/python-coding.pdf>

Windows Terminal Tips



- In windows Terminal you can use the arrow keys



- The up and down arrows can be used to use previous commands
- The left and right arrows can be used to edit a command in place.
- The TAB key can be used for command completion.
- For instance, I want to change to the Documents folder, I can type **cd Do** and then the TAB key and Windows Terminal will complete the command for me.

Python

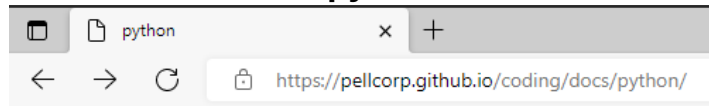


- We are going to learn by doing, including playing with and changing some sample code.

Python Hello World



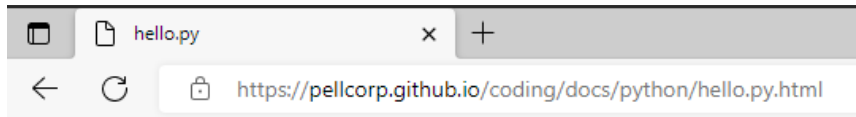
- Open Chrome or Edge
- Type **coding.pellcorp.com** into the location bar and hit the **Enter** key.
- Click on the **hello.py** link



[Back](#)

- [intro](#)
- [server](#)
- [turtle](#)
- [hello.py](#)

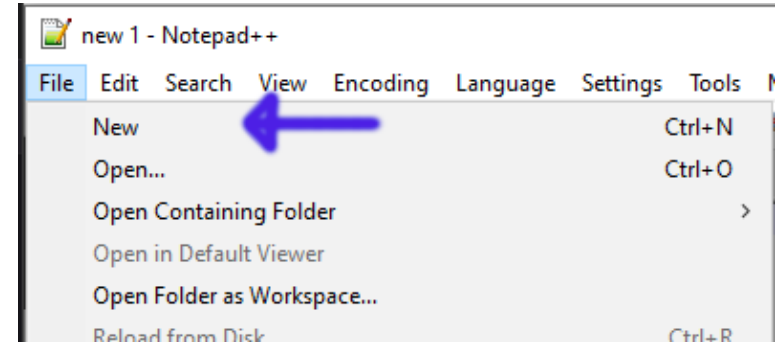
- Select all the code by using **CTRL** key and **A** key



```
print('Hello World')
```

- Copy the selected code by using **CTRL** key and **C** key

- Back to Notepad++
- Use **File** → **New** to open a new tab

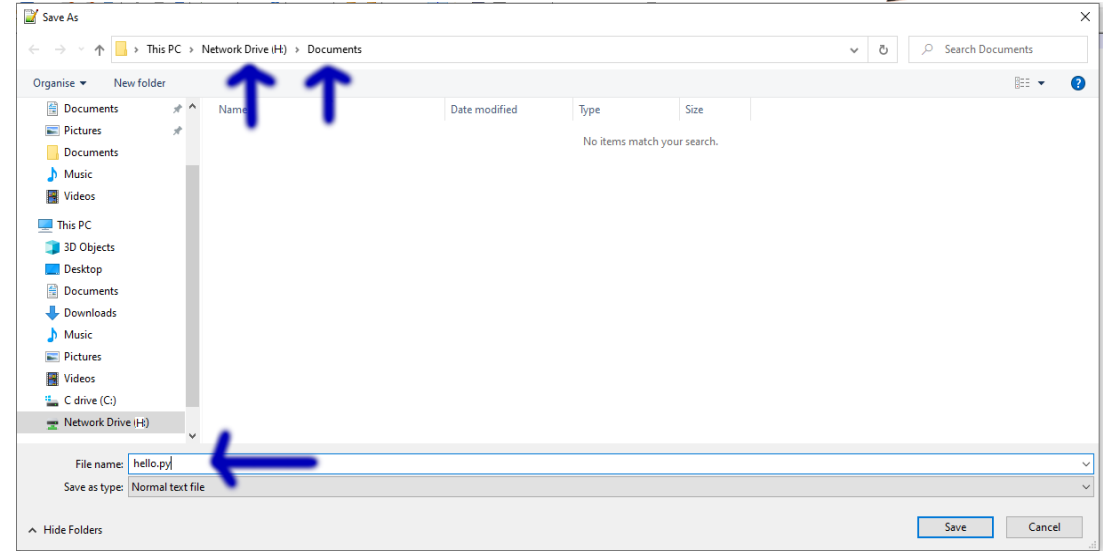


- Paste the code into the new Notepad++ tab using **CTRL** key and **V** key

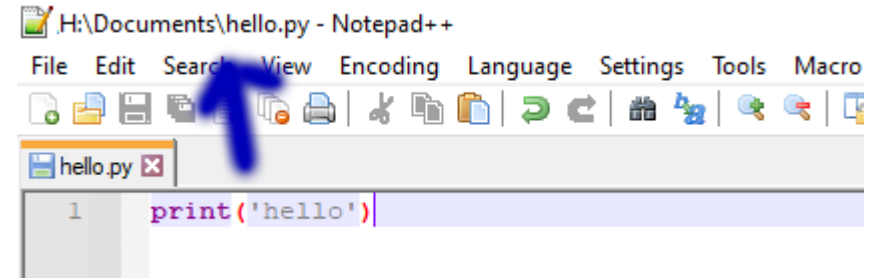
Save hello.py to H:\Documents Cont.



- Click File → Save As



- Save to **H:\Documents** folder with **hello.py** as the File name
- You should now see **H:\Documents\hello.py** at the top of Notepad++



Windows Terminal



- **Windows** Key + **R** Key
- Then **cmd** and **Enter** key
- Change the drive letter to H: by typing **H:** and **Enter** key.
- Change to H:\Documents by typing **cd Documents** and **Enter** key

```
Command Prompt
Microsoft Windows [Version 10.0.19045.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\user>H:

H:\>cd Documents

H:\Documents>_
```

Run hello.py



"C:\Program Files\Python37\python.exe" hello.py

 Command Prompt

```
Microsoft Windows [Version 10.0.19045.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\user>z:

H:\>cd Documents

H:\Documents>"C:\Program Files\Python37\python.exe" hello.py
hello

H:\Documents>_
```

Exercises

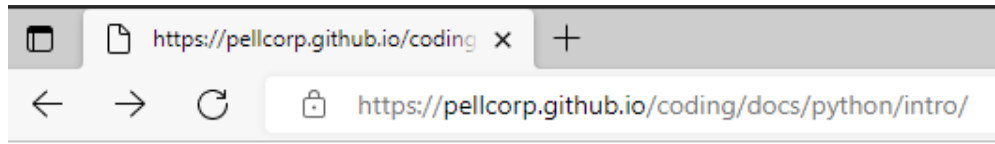


- Run and modify **hello.py**
 - Change the text that is printed from 'hello' to whatever you want, just be sure to keep the single quotes!
 - See what happens if you remove the quotes

Python Conditions

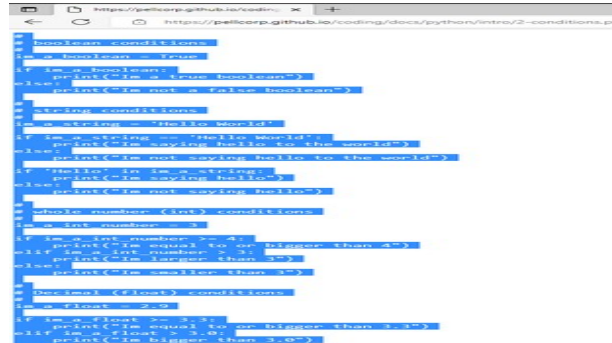


- Open Chrome or Edge
- Type **coding.pellcorp.com** into the location bar and hit the **Enter** key.
- Click on the **2-conditions.py** link

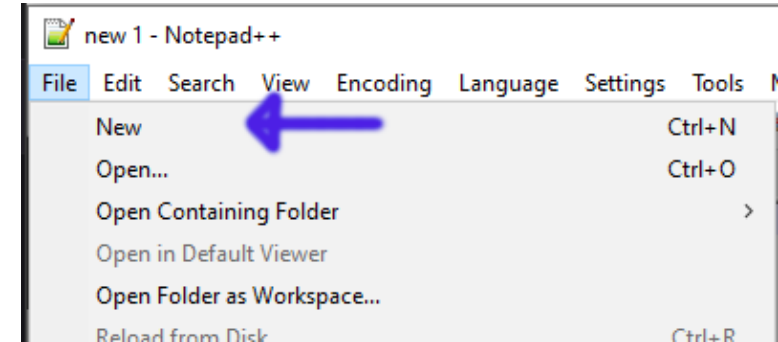


Back

- [1-variables.py](#)
- [2-conditions.py](#)
- [3-list-loop.py](#)



- Back to Notepad++
- Use **File** → **New** to open a new tab



Paste the code into the new Notepad++ tab using **CTRL** key and **V** key

-
-
- Select all the code by using **CTRL** key and **A** key
- Copy the selected code by using **CTRL** key and **C** key

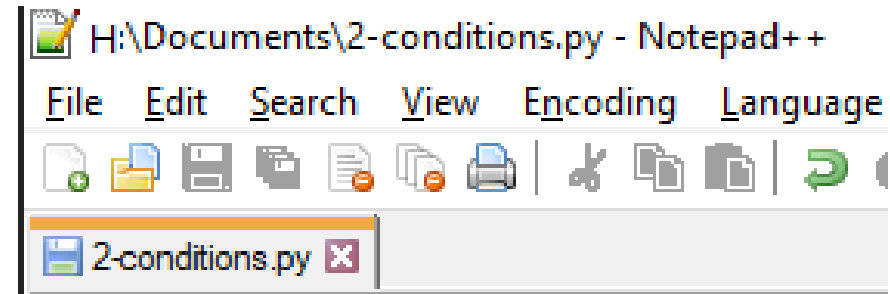
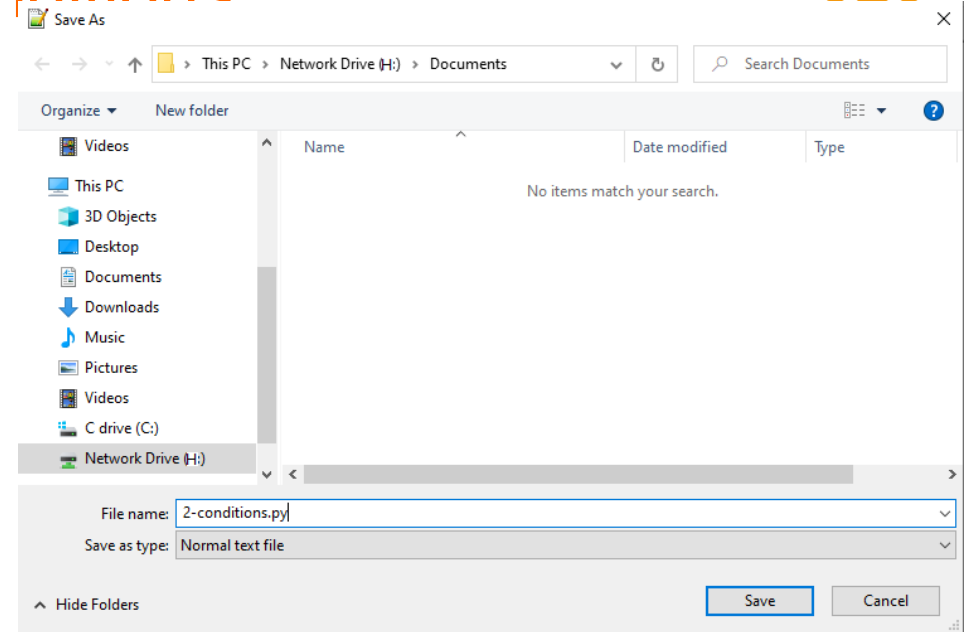
Python Conditions



- Click File → Save As



- Save to **H:\Documents** folder with **2-conditions.py** as the File name
- You should now see **H:\Documents\2-conditions.py** at the top of Notepad++



Run 2-conditions.py



"C:\Program Files\Python37\python.exe" 2-conditions.py

```
Command Prompt

C:\Users\user>H:

H:\>cd Documents

H:\Documents>"C:\Program Files\Python37\python.exe" 2-conditions.py
I am a boolean and I am True
I am a string and I am equal to 'Hello World'
I am a string and I have the 'Hello'
I am a int and I am smaller than 3
I am a float and I am smaller than 3.0
Im a list and I have the word 'jason'
Im a list and I do not have the word 'thor'

H:\Documents>
```

If Statement



```
#  
# string conditions  
#  
im_a_string = 'Hello World'  
  
if im_a_string == 'Hello World':  
    print("I am a string and I am equal to 'Hello World'")  
else:  
    print("I am a string and I am not equal to 'Hello World'")  
  
if 'Hello' in im_a_string:  
    print("I am a string and I have the 'Hello'")  
else:  
    print("I am a string and I no not have the 'Hello'")
```

← *Comment*

← *Equals Comparison*

← *Partial Comparison*

- Condition **operators** include:

- **==** equals
- **!=** not equals
- **>** greater than
- **>=** greater than or equal
- **<** less than
- **<=** less than or equal
- **in** value in a string
- **not in** value not in a string

- Some of the conditions you can use make more sense for comparing numbers than comparing strings.
- For instance **>=** or **<=** and so on

Exercises



- Run and modify **2-conditions.py**
 - For each of the boolean, string, int and float if statements change the `im_a_` variable so that the a different print statement is executed
 - Find the bug(s)