

First Step to Python Programming

Week-1 (How to Use Python: Your First Steps)

Spring Semester, 2022-2023

Download [DOC](#), [SLIDE](#), [PPTX](#)

Outline

- What Python is and why you should use it
- How to Download and Install Python
- The Basic Python Syntax

First Step to Python Programming

First Step to Python Programming

- **Why You Should Use Python**

Python, named after the British comedy group Monty Python, is a high-level, interpreted, interactive, and object-oriented programming language. Its flexibility allows you to do many things, both big and small. With Python, you can write basic programs and scripts and also to create complex and large-scale enterprise solutions

- You can find Python everywhere in the world of computer programming. For example, Python is the foundation of some of the world's most popular websites, including Reddit, Dropbox, and YouTube, to name a few. The Python web framework Django powers both Instagram and Pinterest.

- Compared to other programming languages, Python has the following features:

Interpreted: It's portable and quicker to experiment with than compiled languages.

Multiparadigm: It lets you write code in different styles, including object-oriented, imperative, and functional style.

Dynamically typed: It checks variable types at runtime, so you don't need to declare them explicitly.

Strongly typed: It won't let unsafe operations on incompatible types go unnoticed.



```
34 self.logos = ...
35 self.debug = debug
36 self.logger = logging.getLogger(__name__)
37 if path:
38     self.file = open(os.path.join(path, "requests.log"), "a")
39     self.file.seek(0)
40     self.fingerprints.update({e.request for e in ...})
41
42 @classmethod
43 def from_settings(cls, settings):
44     debug = settings.getbool("DEBUG")
45     return cls(job_dir(settings), debug)
46
47 def request_seen(self, request):
48     fp = self.request_fingerprint(request)
49     if fp in self.fingerprints:
50         return True
51     self.fingerprints.add(fp)
52     if self.file:
53         self.file.write(fp + os.linesep)
54
55 def request_fingerprint(self, request):
56     return request_fingerprint(request)
```

There's a lot more to learn about Python. But by now, you should have a better idea of why Python is so popular and why you should consider learning to program with it.

- **How to Download and Install Python:**

Python works on Linux, Mac, Windows, and several other platforms. It comes preinstalled on macOS and on most Linux distributions. However, if you want to be up to date, then you probably need to download and install the latest version.



Shell

```
$ python3 -V
```

To check what Python version has been installed globally in your operating system, open the terminal or command line and run the following command:

This command prints the version of your system's default Python 3 installation. Note that you use `python3` instead of `python` because some operating systems still include Python 2 as their default Python installation.

檔案總管

csv_process.py

Database.py

config.ini

config.py



已開啟的編輯器

csv_process.py lib\csv 1, M

Database.py lib\db U

config.ini lib\config U

config.py lib\config U

CLIMATE-CRAWLER

lib

lib

lib

lib

lib

lib

lib

lib

lib

lib

lib

lib

lib

lib

lib > db > Database.py

```
1 from sqlalchemy import create_engine
2 from pandas.io.sql import SQLiteDatabase, SQLTable
3 import pandas as pd
4 import numpy as np
5
6 from lib.config.Config import Config
7 from lib.csv.csv_process import load_csv
8
9 class Database:
10     def __init__(self):
11         self.host = 'localhost'
12         self.user_id = 'root'
13         self.password = 'password'
14
15         self.sql_engine = self.create_engine()
16
17         # return: host_ip, user_id
18         # e.g. host_ip = '192.168.191.130:1433', user_id = 'Test_DB'
19     def set_db_config(self):
20         db_config_file_path = 'lib/db/db_config.csv'
21         db_config = pd.read_csv(db_config_file_path)
22         return list(db_config.iloc[0])
```

python

- **The Basic Python Syntax:**
The Python syntax is clear, concise, and focused on readability. Readability is arguably one of the more attractive features of the language itself. It makes Python ideal for people who are learning to program. In this section, you'll learn about several important components of the Python syntax:

```
from lxml import html
from io import StringIO
from io import BytesIO
from colorama import init
from selenium import webdriver
from colorama import Fore, Back, Style
# python cci_events.py 2018-09-01 2018-09-14

def scripter(start_date, end_date):
    print("Current date & time " + time.strftime("%c") + Fore.GREEN + Style.BRIGHT + " Scripter Started")

    filename = '/Users/fagirtmil/Documents/UNCC/DA/SERP/Event_Scripter/datasets/students.csv'
    data = pd.read_csv(filename)
    df = data.groupby(['Event', 'Date'])['Email'].size().reset_index(name='# of Students') # Get total number of students at each event

    # Plot BarChart
    fig, ax = plt.subplots()
    fig.set_size_inches(35.0, 20.0)
    sns.barplot(x='# of Students', y='Event', data=df, ax=ax)
    plt.xticks(size=20)
    plt.yticks(size=20)
    plt.title(start_date + ' to ' + end_date, fontsize=30);
    plt.ylabel('')
    plt.xlabel('# of Students', fontsize=30)
    plt.tight_layout()

    for i, v in enumerate(df['# of Students']):
        ax.text(v + .25, i, str(v), color='black', fontweight='bold', fontsize=20)

    plt.savefig("report.png")
    format = "png"
    bio = BytesIO()
    plt.savefig(bio, format=format)
    print("Content-Type: image/%s\n" % format)

    print("Current date & time " + time.strftime("%c") + Fore.GREEN + Style.BRIGHT + " Scripter Complete")

if __name__ == '__main__':
    init(autoreset=True)
    colorama.init()
    try:
        start_date = sys.argv[1]
        end_date = sys.argv[2]
    except IndexError:
        sys.exit(Fore.YELLOW + Style.BRIGHT + "Either start_date or end_date were not provided, please provide the date range.")

    scripter(start_date, end_date) # Run the Script
```

Python

```
# This is a comment on its own line
```

- **COMMENTS:**

Comments are pieces of text that live in your code but are ignored by the Python interpreter as it executes the code. You can use comments to describe the code so that you and other developers can quickly understand what the code does or why the code is written in a given way. To write a comment in Python, just add a hash mark (#) before your comment text:

SEE YOU NEXT LESSON :)

References

- <https://realpython.com/python-first-steps/#the-basic-python-syntax>
- <https://www.techinside.com/en-populer-programlama-dili-python-oldu/>
- <https://www.abakuskitap.com/blog/icerik/python-ile-ne-yapabilirsiniz-pythonin-3-temel-kullanim-alani>
- <https://www.fiverr.com/fatihagirtmis/do-python-programming-for-you>

End – Of – Week – 1 – Module