**SE-Assignment 6 Introduction to Python**

Python is a high-level, interpreted programming language known for its simplicity, readability, and versatility.

It emphasizes code readability and syntax that allows programmers to express concepts in fewer lines of code compared to languages such as C++ or Java.

**Key Features of Python**

1. Simple and Readable Syntax: Python's syntax is clear and intuitive, making it easy to learn and use, especially for beginners. For example, Python uses indentation to define code blocks, enhancing readability.

for i in range(5):

print(i)

2. Interpreted Language: Python is executed line-by-line, which makes debugging easier and allows for quick testing of code snippets.

3. Dynamic Typing: Variables in Python are dynamically typed, meaning you don't have to declare their type explicitly. The type is determined at runtime.

t = 13 # Integer

s = "Hey" # String

4. High-Level Data Structures: Python provides built-in data structures such as lists, dictionaries, sets, and tuples, which make it easy to work with complex data.

animals = ["cat", "dog", "chicken"]

5. Portability: Python runs on various platforms, including Windows, MacOS, Linux, and Unix, without requiring any changes to the code.

6. Extensive Standard Library: Python's standard library is vast and provides modules and functions for various tasks, including file I/O, system calls, and even web development.

**Use Cases Where Python is Particularly Effective:**

1. Data Science and Machine Learning: Libraries like NumPy, pandas, Matplotlib, and Scikit-learn are widely used for data analysis and machine learning.

import pandas as pd

import numpy as np

dataset = pd.read\_csv('data.csv')

print(dataset.head())

2. Automation and Scripting: Python is often used for writing scripts to automate repetitive tasks.

import os

for filename in os.listdir('.'):

if filename.endswith('.txt'):

print(filename)

3. Game Development: Libraries such as Pygame provide tools for game development.

import pygame

pygame.init()

screen = pygame.display.set\_mode((800, 600))

4. GUI Development: Tkinter and PyQt are commonly used for developing desktop applications with graphical user interfaces.

import tkinter as tk

window = tk.Tk()

window.title("Hello Tkinter")

window.mainloop()

5. Web Development: Frameworks like Django and Flask make it easy to develop robust web applications quickly.

from flask import Flask

app = Flask(\_\_name\_\_)

@app.route('/')

def hello\_world():

return 'Hello, World!'

6. Scientific Computing: Libraries like SciPy and SymPy are used for scientific and mathematical computations.

import numpy as np

x = np.linspace(0, 2 \* np.pi, 100)

y = np.sin(x)

**Describe the steps to install Python on your operating system.**

* Open a web browser and navigate to the Downloads for Windows section of the official Python website. Locate the desired Python version.



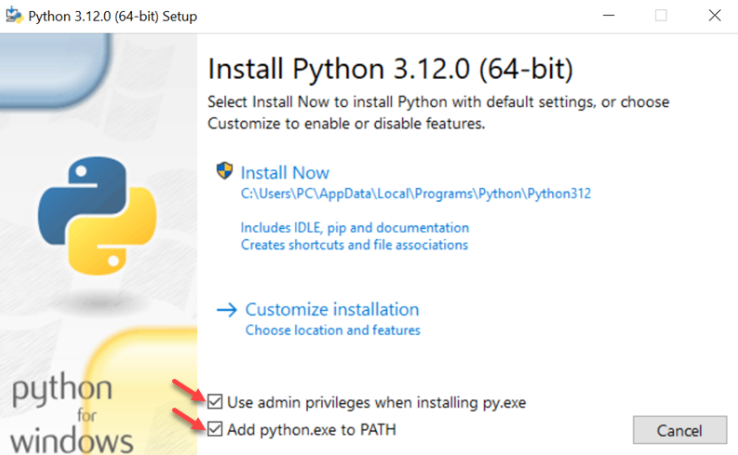
* Click the link to download the file. Choose either the Windows 32-bit or 64-bit installer.

The download is approximately 25MB.

* Run the downloaded Python Installer.

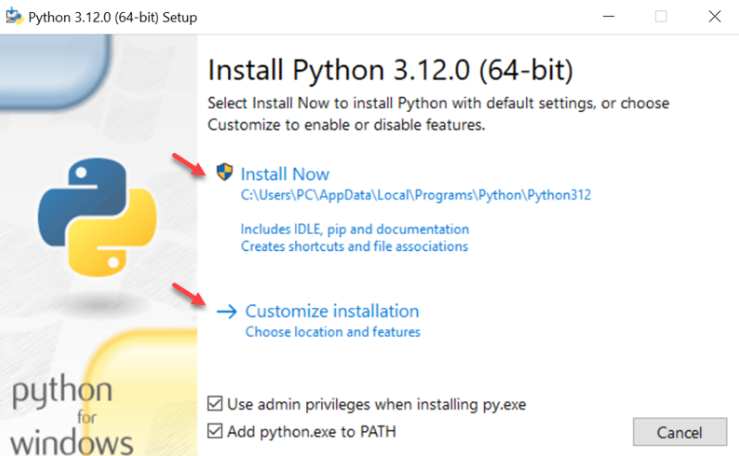
The installation window shows two checkboxes:

* Admin privileges. The parameter controls whether to install Python for the current or all system users. This option allows you to change the installation folder for Python.
* Add Python to PATH. The second option places the executable in the PATH variable after installation. You can also add Python to the PATH environment variable manually later.

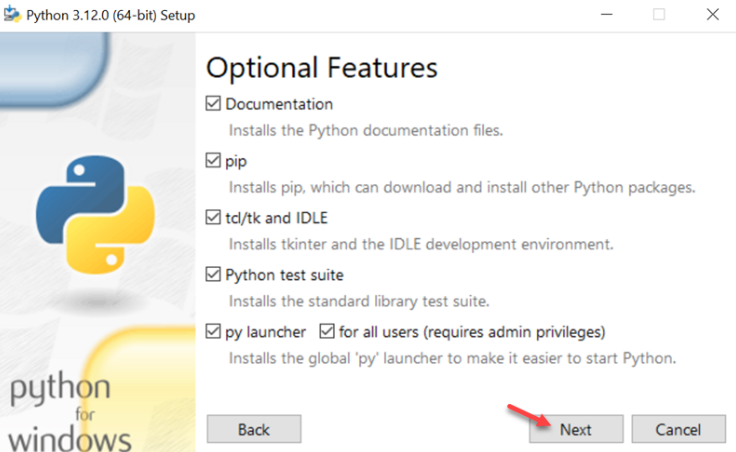


* Select the Install Now option for the recommended installation (in that case, skip the next two steps).

To adjust the default installation options, choose Customize installation instead and proceed to the following step.



* Choose the optional installation features. Python works without these features. Click Next



* Customizing the installation includes advanced options by creating shortcuts, file associations, and adding Python to PATH. After picking the appropriate options, click Install to start the installation.

