3. Python Modules, PIP and Conda

```
- By Srikanth Pinisetti ( Data Engineer )
```

Modules:

Modules in Python act as code libraries, allowing us to utilize pre-written code in our programs. There are two main types:

1. Built-in Modules:
   These modules come bundled with the Python interpreter, so you can use them right away without needing to install anything extra. Examples include:
   - `math`: for mathematical functions like `sqrt()` and `sin()`
   - `datetime`: for working with dates and times
   - `random`: for generating random numbers
   - `os`: for interacting with the operating system

```
import math
print(math.sqrt(16))  # Output: 4.0

import datetime
print(datetime.datetime.now())  # Output: Current date and time

import random
print(random.randint(1, 10))  # Output: Random integer between 1 and 10
```

2. External Modules:
   These modules are created by third-party developers and need to be installed separately, often using a package manager like PIP or Conda. This allows us to access a wide range of additional functionality. Examples include:
   - `requests`: for making HTTP requests
   - `matplotlib`: for creating plots and graphs
   - `pandas`: for data manipulation and analysis

```
pip install requests
pip install matplotlib
pip install pandas
```

```
import requests
response = requests.get("https://www.example.com")
print(response.status_code)  # Output: Status code of the HTTP response

import matplotlib.pyplot as plt
```

```
plt.plot([1, 2, 3, 4], [1, 4, 9, 16])
plt.show()  # Display the plot

import pandas as pd
data = {'Name': ['Alice', 'Bob', 'Charlie'], 'Age': [25, 30, 35]}
df = pd.DataFrame(data)
print(df)  # Output: DataFrame containing the data
```

PIP:

PIP (Python Package Installer) is a package manager for Python that helps you find, install, and manage external modules easily. You can use it from the command line to install packages from the Python Package Index (PyPI) or other sources.

Conda:

Similarly we have Conda: Another package manager, primarily used within the context of the Anaconda distribution, which is a popular Python and R data science platform. While PIP is Python-specific, Conda is a more general package manager that can handle not only Python packages but also packages from other languages like R, Ruby, Lua, Scala, Java, JavaScript, C/C++, FORTRAN, and more.