Python Basics Cheatsheet for Engineers

---

1. Setting Up Python on Windows

1. **Install Python**:
   - Download from python.org.
   - During installation, check **"Add Python to PATH"**.
2. **Verify Installation**:
   - Open Command Prompt and type:
     `python --version`
3. **Install pip (Python Package Manager)**:
   - Check if pip is installed:
     `pip --version`
   - If not, run:
     `python -m ensurepip --upgrade`

---

2. Setting Up a Virtual Environment

1. **Create a Virtual Environment**:
   - Navigate to your project folder in Command Prompt:

     ```
     cd path\to\your\project
     ```

   - Create the virtual environment:

     ```
     python -m venv venv
     ```

2. **Activate the Virtual Environment**:

   ```
   venv\Scripts\activate
   ```

   - You will see `(venv)` before the command prompt.
3. **Deactivate the Virtual Environment**:

   ```
   deactivate
   ```

4. **Install Required Packages**:
   - Install packages like numpy or pandas:

     ```
     pip install numpy pandas
     ```

   - Install packages with a specific version:

     ```
     pip install numpy==1.19.3
     ```

   - Install all packages from a requirements file:

     ```
     pip install -r requirements.txt
     ```

- Create a requirements file:

  ```
  pip freeze > requirements.txt
  ```

- Uninstall a package:

  ```
  pip uninstall package_name
  ```

- List installed packages:

  ```
  pip list
  ```

---

3. Setting Up VS Code

1. **Install VS Code**:
   - Download from code.visualstudio.com.
2. **Install Extensions**:
   - Open VS Code and go to the Extensions view (`Ctrl+Shift+X`).
   - Install:
     - **Python** (by Microsoft)
     - **Jupyter**
     - **GitHub Copilot** (if available)
3. **Set Python Interpreter**:
   - Press `Ctrl+Shift+P` > Select **Python: Select Interpreter**.
   - Choose your virtual environment from the list.
4. **Create a Jupyter Notebook**:
   - In VS Code, create a new file with `.ipynb` extension.
   - Click **"Run Cell"** to execute code in the notebook.

---

4. Basic Python Commands

Math Operations:

```
# Basic math
x = 10 + 5  # Addition
x = 10 - 5  # Subtraction
x = 10 * 5  # Multiplication
x = 10 / 2  # Division
x = 10 ** 2  # Power

# Import math library
import math
math.sqrt(16)  # Square root
math.pi        # Value of pi
```

Data Types:

```
# Numbers
x = 10        # Integer
y = 10.5      # Float
z = 10 + 5j   # Complex
print(type(x), type(y), type(z))
# Output: <class 'int'> <class 'float'> <class 'complex'>

# Strings
name = "John"
print(name.upper())  # Uppercase
print(name.lower())  # Lowercase

# Lists
numbers = [1, 2, 3, 4]
numbers.append(5)  # Add to list
print(numbers)

# Dictionaries
person = {"name": "John", "age": 30}
print(person["name"])
```

Loops:

```
# For loop
for i in range(5):
    print(i)

# While loop
count = 0
while count < 5:
    print(count)
    count += 1
```

Functions:

```
def greet(name):
    return f"Hello, {name}!"

print(greet("Alice"))
```

---

5. Basic Data Analysis with Pandas

1. **Import Libraries**:

```
import pandas as pd
import numpy as np
```

2. **Create a DataFrame**:

```
data = {
    "Name": ["Alice", "Bob", "Charlie"],
    "Age": [25, 30, 35],
    "Salary": [50000, 60000, 70000]
}
df = pd.DataFrame(data)
print(df)
```

3. **Read/Write CSV Files**:

```
# Read a CSV file
df = pd.read_csv("data.csv")

# Write to a CSV file
df.to_csv("output.csv", index=False)
```

4. **Basic Operations**:

```
print(df.head())         # First 5 rows
print(df.describe())     # Summary statistics
print(df["Age"].mean())  # Average age
```

---

6. Using Copilot for Assistance

1. **Enable Copilot**:
   - Ensure GitHub Copilot is installed in VS Code.
   - Start typing, and suggestions will appear automatically.
2. **Example Use Case**:
   - Type:
     ```
     # Calculate the average of a list
     def calculate_average(numbers):
     ```
   - Copilot will generate code suggestions.

---

7. Troubleshooting

1. **Common Errors**:
   - `ModuleNotFoundError`: Install the missing package:
     `pip install package_name`
   - `SyntaxError`: Check for typos or indentation issues.
2. **Restart Jupyter Kernel**:
   - If code behaves unexpectedly, restart the kernel in VS Code.

---

8. Best Practices

1. **Write Clean Code**:
   - Use meaningful variable names.
   - Write comments to explain your code.
2. **Save Work Regularly**:
   - Use version control (Git) to track changes.
3. **Ask for Help**:
   - Use forums like Stack Overflow or read Python documentation.

This cheatsheet covers the essentials to get started with Python, data analysis, and development using VS Code. Keep it handy for quick reference!