Wexdi SoftSkills

Introduction to Python Programming

Understanding Python basics

Wexdi SoftSkills

- Python is a versatile and widely-used programming language known for its simplicity and readability.
- It is an excellent choice for beginners and experienced programmers alike.

Understanding Python basics

Wexdi SoftSkills

- **Writing Your First Python Program**
  - To get started with Python, you can write a simple program that prints "Hello, World!" to the console.
  - Here's an example:

```
print("Hello, World!")
```

  - This line of code uses the `print()` function to display the text on the screen

Understanding Python basics

- **Variables and Data Types**
  - In Python, variables are used to store values.
  - You can assign values to variables using the assignment operator =. Python supports various data types, including integers, floating-point numbers, strings, lists, tuples, dictionaries, and sets.
  - Here are some examples:
    - ```
      # Integer
      age = 25

      # Floating-point number
      salary = 50000.75
      ```
      ```

Understanding Python basics

- **Control Flow**
  - Python provides several control flow statements to manage the execution of code based on conditions.
  - The most common ones are `if`, `elif`, and `else` statements.
  - Here's an example:

```
x = 10
if x > 0:
        print("x is positive")
elif x == 0:
        print("x is zero")
else:
        print("x is negative")
```

  - You can also use loops to iterate over sequences. The `for` loop and `while` loop are commonly used for this purpose

Understanding Python basics

Wexdi SoftSkills

- **Functions**
  - Functions in Python are defined using the `def` keyword. They allow you to encapsulate reusable blocks of code. Here's an example of a simple function:

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

  - Functions can take parameters and return values. They help in organizing and modularizing your code

Understanding Python basics

- **Lists and Dictionaries**
  - Lists and dictionaries are essential data structures in Python. Lists are ordered collections of items, while dictionaries store key-value pairs.
  - Here are some examples:

```
# List
    numbers = [1, 2, 3, 4, 5]
    print(numbers[0]) # Output: 1

# Dictionary
    student = {"name": "Bob", "age": 22}
    print(student["name"]) # Output: Bob
```

  - You can perform various operations on lists and dictionaries, such as adding, removing, and modifying elements

Understanding Python basics

- **Exception Handling**
  - Python provides a way to handle exceptions (errors) using the `try`, `except`, `else`, and `finally` blocks. This helps in managing errors gracefully.
  - Here's an example:

```
try:
result = 10 / 0
except ZeroDivisionError:
print("Error: Division by zero")
else:
print("No errors occurred")
finally:
print("This block always executes")
```

  - Exception handling ensures that your program can handle unexpected situations without crashing

Setting up your environment

- **To set up a Python environment, you need to**:
  - Install Python on your machine, either from the official website or using a package manager like Chocolatey for Windows.
  - Create a virtual environment for your project, using the command python -m venv /path/to/new/virtual/environment or python3 -m venv /path/to/new/virtual/environment depending on your Python version.
  - Activate the virtual environment, using the command source env/bin/activate for Unix/macOS or env\Scripts\activate for Windows.
  - Install the required packages for your project, using the command pip install -r requirements.txt or pipenv install if you have a Pipfile.
  - Select and configure an interpreter for your code editor, using the Python: Select Interpreter command from the Command Palette in VS Code or the equivalent in other editors.

Writing your first Python script,

To write your first Python script, you can follow these steps:

1. Ensure that you have the necessary prerequisites.
2. Create a Python project. If you're on the Welcome screen, click New Project.
3. Create a Python file. In the Project tool window, select the project root, right-click it, and select File | New.
4. Edit Python code.
5. Run your application.