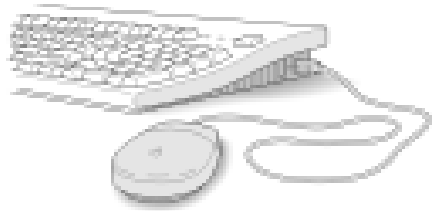
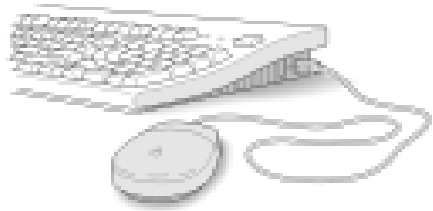
Basic I/O in Python

Basic Program Structure

Basic Program Structure

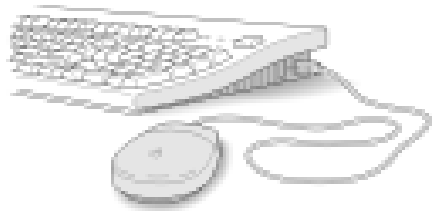


Basic Program Structure



```
x = (input / 56.7) ** 2.5
y = x + 57.0 % 6
z = (x + y) / 2.0
z = z + 4 * x ** 2
```

Basic Program Structure



```
x = (input / 56.7) ** 2.5
y = x + 57.0 % 6
z = (x + y) / 2.0
z = z + 4 * x ** 2
```

Basic I/O Funcitons

Basic I/O Funcitons

- `input()` – prompt user for input from terminal

Basic I/O Funcitons

- `input()` – prompt user for input from terminal
- `print()` – display program output to terminal window

Basic I/O Funcitons

- `input()` – prompt user for input from terminal
- `print()` – display program output to terminal window

```
userInput = input("Enter your input here:  ")
```

Basic I/O Funcitons

- `input()` – prompt user for input from terminal
- `print()` – display program output to terminal window

```
userInput = input("Enter your input here:  ")
# Do some calculations with userInput variable
...
```

Basic I/O Funcitons

- `input()` – prompt user for input from terminal
- `print()` – display program output to terminal window

```
userInput = input("Enter your input here:  ")
# Do some calculations with userInput variable
...
outputMessage = ...
```

Basic I/O Funcitons

- `input()` – prompt user for input from terminal
- `print()` – display program output to terminal window

```
userInput = input("Enter your input here:  ")
# Do some calculations with userInput variable
...
outputMessage = ...
print(outputMessage)
```

The input() function

The input() function

Basic form:

```
stringInput = input("Please type your input:  ")
```

The input() function

Basic form:

```
stringInput = input("Please type your input:  ")
```

For numerical inputs:

```
floatInput = float(input("Enter any number:  "))
```

```
intInput = int(input("Enter an int:  "))
```

Concept Check!

What is the value (and type) of `userInput` after each of the following commands/user inputs?

1.
```
>>> userInput = input()
5
```
2.
```
>>> userInput = input("Enter a number:  ")
Enter a number:  5
```
3.
```
>>> userInput = int(input("Enter a number:  "))
Enter a number:  5
```
4.
```
>>> userInput = int(input("Enter a number:  "))
Enter a number:  5.0
```

The print() function

The print() function

Basic Functionality:

```
print("The answer is:")
print(variableName)
```

The print() function

Basic Functionality:

```
print("The answer is:")
print(variableName)
```

No New Lines:

```
print("The answer is:  ", end="")
print(variableName)
```

The print() function

Basic Functionality:

```
print("The answer is:")
print(variableName)
```

No New Lines:

```
print("The answer is:  ", end="")
print(variableName)
```

Print multiple items:

```
print("The answer is:  ", variableName)
```

or

```
print("The answer is:  " + str(variableName))
```

Basic String Tricks

Basic String Tricks

Triple Quotes:

```
""" This text will
display on two lines"""
```

Basic String Tricks

Triple Quotes:

```
""" This text will
display on two lines"""
```

Special Formatting Characters:

"\n" – new line

"\t" – tab

f-strings

f-strings

Basic f-string usage:

```
myVar = 5
print(f"My variable has the value:  {myVar}")
```

f-strings

Basic f-string usage:

```
myVar = 5
print(f"My variable has the value:  {myVar}")
```

Advanced f-string usage:

```
myVar = 5
print(f"My variable is less than:  {myVar + 1}")

myVar = 1.23456
print(f"My variable has the value:  {myVar:.2f}")
```

f-strings

Basic f-string usage:

```
myVar = 5
print(f"My variable has the value:  {myVar}")
```

Advanced f-string usage:

```
myVar = 5
print(f"My variable is less than:  {myVar + 1}")

myVar = 1.23456
print(f"My variable has the value:  {myVar:.2f}")
```

Learn more:

https://docs.python.org/3/library/string.html#formatspec

Concept Check!

Let `myFloat = 1.499999` and `myInt = 1`.
What will the following statements output?

1. `print(myFloat)`
2. `print("The float is:  " + str(myFloat))`
3. `print("The int is:  ", myInt)`
4. `print("The int is:  ", end="")`
   `print(myInt)`
5. `print("""The int has the value`
   `in myInt and`
   `the float has the value in myFloat""")`
6. `print(f"The int is less than:  {myInt + 1}")`
7. `print(f"The float is about:  {myFloat:.1f}")`