

## Week 1 - Python Basics / 1. Introduction to Python

### Exercises 1

---

1. What is the different between an Algorithm and a Program?

Answer: An algorithm is a conceptual idea; a program is a concrete instantiation of an algorithm.

2. True or False? A computational mode of thinking means that everything can be viewed as a math problem involving numbers and formulas.

Answer: True

3. True or False? Computer Science is the study of how to build efficient machines that run programs

Answer: False

4. The two things every computer can do are:

Answer: Perform calculations, Remember the results

### Exercises 2

---

1. Declarative knowledge refers to statements of fact

Answer: True

2. Imperative knowledge refers to 'how to' methods.

Answer: True

3. A recipe for deducing the square root involves guessing a starting value for y. Without another recipe to be told how to pick a starting number, the computer cannot generate one on its own.

Answer: True

### Exercises 3

---

1. True or False? A stored program computer is designed to compute precisely one computation, such as a square root, or the trajectory of a missile.

Answer: False

2. True or False? A fixed program computer is designed to run any computation, by interpreting a sequence of program instructions that are read into it.

Answer: False

3. A program counter

Answer: Points the computer to the next instruction to execute the program

4. What does it mean when we say that "the computer walks through the sequence executing some computation"?

Answer: The computer executes the instructions mostly in a linear sequence, except sometimes it jumps to a different place in the sequence.

5. True or False? In order to compute everything that is computable, every computer must be able to handle the sixteen most primitive operations.

Answer: False

## Exercises 4

---

1. Determines whether a string is legal

Answer: Syntax

2. Determines whether a string has meaning

Answer: Static semantics

3. Assigns a meaning to a legal sentence

Answer: Semantics

## Exercises 5

---

For each of the following expressions, indicate the type of the expression. While you could simply type these expressions into your shell, we encourage you to answer them directly since this will help reinforce your understanding of basic Python expressions.

3.14

Answer: float

-34

Answer: int

True

Answer: bool

None

Answer: NoneType

3.0

Answer: float

## Exercises 6

---

For each of the following expressions, indicate the value returned, or if the evaluation would lead to an error, write the word 'error' (note this is a word, not a string, no quotes). While you could simply type these expressions into an IDE, we encourage you to answer them directly since this will help reinforce your understanding of basic Python expressions.

For decimal answers, give the full result, or four decimal places of accuracy (whichever is shortest).

$6 + 12 - 3$

Answer: 15

$2 * 3.0$

Answer: 6.0

$-- 4$

Answer: 4

10/3

Answer: 3.3333

10.0/3.0

Answer: 3.3333

(2 + 3) \* 4

Answer: 20

2 + 3 \* 4

Answer: 14

2 \*\* 3 + 1

Answer: 9

2.1 \*\* 2.0

Answer: 4.41

2.2 \* 3.0

Answer: 6.6

## Exercises 7

---

Below is a transcript of a session with the Python shell. For each expression being evaluated, provide the type and the value the expression returns. If evaluating an expression would cause an error, select 'NoneType' and write the word 'error' (note this is a word, not a string, no quotes) in the box. While you could simply type these expressions into your IDE, we encourage you to answer them directly since this will help reinforce your understanding of basic Python expressions.

Assume that the expressions are evaluated in the order shown – that is, each problem part is evaluated directly after the previous problem part(s).

1.

```
>>> a = 3
>>> a + 2.0
```

Answer: float

Answer: 5.0

2.

```
>>> a = a + 1.0
>>> a
```

Answer: float

Answer: 4.0

3.

```
>>> a = 3
>>> b
```

Answer: NoneType

Answer: error

## Exercises 8

---

For each of the following expressions, indicate the value returned, or if the evaluation would lead to an error, write the word 'error' (note this is a word, not a string, no quotes). While you could simply type these expressions into your IDE, we encourage you to answer them directly since this will help reinforce your understanding of basic Python expressions.

`3 > 4`

Answer: False

`4.0 > 3.999`

Answer: True

`4 > 4`

Answer: False

`4 > + 4`

Answer: False

`2 + 2 == 4`

Answer: True

`True or False`

Answer: True

`False or False`

Answer: False

`Not False`

Answer: True

`3.0 - 1.0 != 5.0 - 3.0`

Answer: False

`3 > 4 or (2 < 3 and 9 > 10)`

Answer: False

`4 > 5 or 3 < 4 and 9 > 8`

Answer: True

`Not (4 > 3 and 100 > 6)`

Answer: False

## Exercises 9

---

Below is a transcript of a session with the Python shell. For each expression being evaluated, provide the type and the value that the last expression in the transcript returns. If evaluating an expression would cause an error, select 'NoneType' and write the word 'error' (note this is a word, not a string, no quotes) in the box. While you could simply type these expressions into your IDE, we encourage you to answer them directly since this will help reinforce your understanding of basic Python expressions.

Assume that the expressions are evaluated in order shown – that is, each problem part evaluated directly after the previous problem part(s).

1.

```
>>> a = 3
>>> a == 5.0
>>> a
```

Answer: int

Answer: 3

2.

```
>>> b = 10
>>> c = b > 9
>>> c
```

Answer: bool

Answer: true

## Exercises 10

---

For each of the following expressions, indicate the type of the expression and the value returned, or, if the evaluation would lead to an error, choose the type 'NoneType' and write the word 'error' (note this is a word, not a string, no quotes) as the value returned.

While you could simply type these expressions into your IDE, we encourage you to answer them directly since this will help reinforce your understanding of basic Python expressions.

3 + 5.0

Answer: float

Answer: 8.0

5/2

Answer: float

Answer: 2.5

5/2 == 5/2.0

Answer: bool

Answer: true

5/2.0

Answer: float

Answer: 2.5

Round(2.6)

Answer: int

Answer: 3

Int(2.6)

Answer: int

Answer: 2

2.0 + 5.0

Answer: float

Answer: 7.0

**MITx 6.00.1x**

**Introduction to Computer Science and Programming Using Python**

`5*2 == 5.0 * 2.0`

Answer: bool

Answer: true