Introduction to Programming

Exercises

Week 1

Prior to attempting these exercises ensure you have read the lecture notes and/or viewed the video, and also completed the practical. You may wish to use the Python interpreter in interactive mode to help work out the solutions to some of the questions.

Download and store this document within your own filespace, so the contents can be edited. You will be able to refer to it during the test in Week 6.

Enter your answers directly into the highlighted boxes.

For more information about the module delivery, assessment and feedback please refer to the module within the MyBeckett portal.

What is the name of the programming language that we will be using on this module? What version of the language are we using?

Answer:

Python is the name of the programming language that we are using on this module, We are using **Python 3.7 or later version.**

_____

A computer program takes some *input*, performs some *processing* then…. what?

Answer:

A computer program takes some *input*, performs some *processing* then generate output.

_____

What generation of programming language is *machine code*?

Answer:

The **first** generation of programming language is *machine code.*

_____

Which of the following is known as a second generation programming language?

- C++
- Java
- Assembly
- R
- Python

Answer:

Assembly is known as a second-generation programming language

_____

State one problem associated with writing code in Assembly Language.

Answer:

Different CPUs require different assembly languages. Therefore, code is not portable.

_____

What generation of programming language is *Python*?

Answer:

*Python is the **third** generation of programming language.*

_____

What is the purpose of a *compiler*?

Answer:

The purpose of a compiler is to translate high-level programming code, such as those written in languages like C, C++, or Java into machine or binary code that a computer's processor can execute directly. A compiler processes the entire block of code at once.

_____

The Python interpreter uses an interaction model called **REPL**. What does this stand for?

Answer:

REPL stands for Read, Evaluate, Print, and Loop.

_____

Is it true that Python development always has to take place using *interactive mode* within the Python interpreter?

Answer:

No, Python development does not always have to take place using *interactive mode* within the Python interpreter.

_____

What does the term IDE stand for?

Answer:

IDE stands for Integrated Development Environment.

_____

What is the main reason why programmers use *code libraries*?

Answer:

Code libraries are primarily used by programmers to encourage code reusability by allowing them to utilize pre-written and tested code modules,saving development time and effort.

_____

The Python language is often used in the field of *data science*. What other language specifically supports *data science*?

Answer:

' R ' language specifically supports *data science.*

_____

An expression within a programming language consists of *operands* and *operators*.

Given an expression such as 20 + 10, which part of this is the *operator*?

Answer:

'+ ' is the operator.

And, which part of this is the *operand*?

Answer:

20 and 10 are the operands.

_____

Within Python, what calculation is performed by the '*' operator?

Answer:

Within Python, **multiplication** is performed by the ' * ' operator.

And, what calculation is performed by the '/' operator?

Answer:

Division is performed by the ' / ' operator.

And, what calculation is performed by the '**' operator?

Answer:

> **Exponentiation** is performed by the '**' operator.

Using the information about expression evaluation provided in the related tutorial, evaluate each of the following expressions **in your head** and type the result in the answer boxes below. Remember that an operator precedence is applied, but can be overridden by the use of parentheses.

```
a) 100 + 200 - 50
```

Answer:

250

```
b) 10 + 20 * 10
```

Answer:

210

```
c) 20 % 3
```

Answer:

2

```
d) 20 / (2 * 5)
```

Answer:

2

```
e) 20 / 2 * 5
```

Answer:

50

```
f) 10 * 2 + 1 * 3
```

Answer:

23

g) `5 + 10 ** 2`

Answer:

105

h) `(10 + 2 / 2) + ((10 * 2) ** 2)`

Answer:

411

_____

Use the Python interpreter to input and then execute a simple Python expression that adds the three numbers 100.6, 200.72 and 213.3, then write the result in the answer box below.

Answer:

Result=514.62

_____

Use the Python interpreter to input and then execute a simple Python expression that multiplies the three numbers 20.25, 100 and 23.9, then write the result in the answer box below.

Answer:

Result=48517.5

_____

Use the Python interpreter to input and then execute a simple Python expression that divides the number 10 by 0, then write the result in the answer box below.

Answer:

It shows ZeroDivisionError.

_____

What type of error is typically easier to identify? A *syntax* error? Or a *logical* error?

Answer:

A *syntax* error is typically easier to identify.

_____

What type of message is used by the Python interpreter to report run-time errors?

Answer:

'Traceback' message is used by the Python interpreter to report run-time errors.

_____

What command can be used to exit the Python interpreter?

Answer:

exit() command can be used to exit the Python interpreter.

_____

Exercises are complete

Save this logbook with your answers. Then ask your tutor to check your responses to each question.