## 0. Introduction

### 1. The fundamentals of programming

### 2. Following along with the course

### 3. Prerequisites

## Prerequisites for *Programming Foundations: Fundamentals*

Welcome to *Programming Foundations: Fundamentals*! While this course presents foundational information about coding, you should already be familiar with computers in general and probably with at least a little programming specifically.

[Introduction to Career Skills in Software Development](https://www.linkedin.com/learning/introduction-to-career-skills-in-software-development?u=83641554) may be a useful foundation if you haven’t done any programming in the past. It will give you a sense of what’s involved in working on software development projects, from the very basics to larger works, and includes just a taste of programming that will help get you started here.

## ‘1. Programming Basics

### 1. What is programming

### 2. What is a programming language

### 3. Writing source code

### 4. Running your code

### 5. Using an IDE

### Chapter Quiz

## 2. Programming Syntax

### 1. Why Python

### 2. Installing Python on a Mac

### 3. Installing Python on Windows

### 4. Running Python on the command line on Mac

### 5. Running Python on the command line on Windows

### 6. Installing Visual Studio Code on a Mac

### 7. Installing Visual Studio Code on Windows

### 8. Running Python in an IDE

### 9. Basic statements and expressions

### 10. Troubleshooting issues

### Chapter Quiz

## 3. Variables and Data Types

### 1. Introduction to variables and data types

### 2. Variables across languages

### 3. Working with numbers

### 4. Working with strings

### 5. Properly using whitespace

### 6. Working with comments

### 7. Challenge - What's the output

### 8. Solution - What's the output

### Chapter Quiz

Question 1 of 17

When would a removal of white space be problematic?

removing an empty line between two lines of code

Incorrect

Empty lines inside the code are only used for better readability and do not impact the code itself.

removing the space after the "if" in if x==1

Correct

removing the space after the string in print("Hi, ",name)

Incorrect

removing the spaces around the equal sign in x = 1

Incorrect

Question 2 of 17

Although Python ignores empty lines, one place where whitespace does make a difference is inside of \_\_\_\_\_.

files

Incorrect

functions

Incorrect

strings

Incorrect

special keywords

Correct

Question 3 of 17

Which expression evaluates to 2?

5 // 3

Incorrect

5 % 3

Correct

2 / 4

Incorrect

Question 4 of 17

Python treats all numbers the same, regardless of numerical value.

FALSE

Correct

TRUE

Incorrect

Question 5 of 17

Which string definition results in an error?

'It is 10am'

Incorrect

"It's 10am"

Incorrect

There is no issue with this code, since it correctly uses a single quote inside a double-quoted string.

'It's 10am'

Correct

"It is 10am"

Incorrect

Question 6 of 17

Which of these will not output:

I'm learning a lot!

print('I'm learning a lot!')

Correct

message = "I'm learning a lot!" print(message)

Incorrect

print("I'm learning a lot!")

Incorrect

Question 7 of 17

Which character starts a comment in Python?

?

Incorrect

This character is not used for commenting lines in Python.

/

Incorrect

#

Correct

$

Incorrect

Question 8 of 17

What is the output of the following Python code?

first\_name = "Jeff"

#first\_name = "Sara"

print(first\_name)

JeffSara

Incorrect

Sara

Incorrect

Jeff

Correct

Question 9 of 17

What will the following code print?

number = 3\*\*2

Number = 2\*\*3

print(number)

9

Correct

6

Incorrect

8

Incorrect

Since the lowercase number is printed, the uppercase Number is ignored.

This code will error out.

Incorrect

Question 10 of 17

What is the output of this Python code?

first\_name = "Jeff"

first\_Name = "Sara"

print(first\_name)

Jeff, Sara

Incorrect

Sara

Incorrect

Jeff

Correct

Question 11 of 17

Which variable name is valid in Python?

3blindmice

Incorrect

$money

Incorrect

account\_balance

Correct

Question 12 of 17

Which data type in Python represents a number with a decimal point?

Float

Correct

Frac

Incorrect

Dec

Incorrect

There is no built-in Dec data type in Python.

Int

Incorrect

Question 13 of 17

How would you fix the following code to return the remainder of 1,000 divided by 300?

1,000 % 300

1,000 // 300

Incorrect

1000 % 300

Correct

1000 \*\* 300

Incorrect

1,000 %% 300

Incorrect

The remainder operator does not use two percent symbols.

Question 14 of 17

What will the following code print?

variable = "12"

print(type(variable))

int

Incorrect

Since the variable is defined as a string, its type is not going to be int.

12

Incorrect

twelve

Incorrect

str

Correct

Question 15 of 17

What operator do you use to assign a value to a variable?

plus (+)

Incorrect

equals (=)

Correct

assign (&)

Incorrect

Question 16 of 17

Which data type is a whole number?

bool

Incorrect

int

Correct

strt

Incorrect

num

Incorrect

Question 17 of 17

Which variable name is valid in Python?

max distance

Incorrect

break

Incorrect

1st\_rule

Incorrect

This variable name is not valid, since it starts with a number.

speed\_limit

Correct

## 4. Conditional Code

### 1. Making decisions in code

### 2. Exploring conditional code

### 3. Working with simple conditions

### 4. Conditionals across languages

### 5. Challenge - Guessing game

### 6. Solution - Guessing game

### Chapter Quiz

## 5. Modular Code

### 1. Introduction to functions

### 2. Creating and calling functions

### 3. Setting parameters and arguments

### 4. Returning values from functions

### 5. Functions across languages

### 6. Challenge - Favorite cities

### 7. Solution - Favorite cities

### Chapter Quiz

Question 1 of 15

Which keyword does Python use to define a function?

def

Correct

func

function

sub

Question 2 of 15

What does the void keyword mean when it is specified before a function definition?

The function does not return a value.

Correct

The function name is "void".

The function voids all variables it received.

The function is not expecting any variables.

Question 3 of 15

What can the following function be used for?

def mystery(x):

if (x % 2 == 0):

return("yes")

else:

return("no")

checking whether a variable is a numeric value

checking whether a number has more than 2 digits

Incorrect

checking whether a number is larger than 2

Incorrect

The operator used in this code is not a comparison operator but a remainder operator.

checking whether a number is even or odd

Correct

Question 4 of 15

Which is not a benefit of functions?

With functions, you don't have to repeat the same code over and over again.

Functions eliminate crashes in your code.

Correct

It's easier to fix a mistake in a function than one copied and pasted in multiple places.

Question 5 of 15

What will the following code print?

def compare():

print(5, "is greater than", 6)

5 is greater than 6

This code will not print anything.

Correct

56

is greater than

Question 6 of 15

The code that's contained inside of a function is often called the function's legs.

FALSE

TRUE

Incorrect

Creating and calling functionsReplay

Review this video

Creating and calling functions

3m 47s

Question 7 of 15

Why is there no output for this program?

def goodbye():

print("Bye")

We never called the goodbye function.

Correct

It's so hard to say goodbye.

We didn't spell goodbye properly.

Question 8 of 15

Given the following code, how would you call the function?

def hello(name):

print("Hey,", name)

hello() "John"

"John">hello()

Incorrect

name("John")

hello("John")

Correct

Question 9 of 15

What is the issue with this function as defined?

def double():

print(x\*2)

The function name must start with a capital letter.

The number to double is not received.

Correct

A numeric value must be converted to string before printing.

The function must return a value.

Question 10 of 15

Which Python keyword is used to send back values from a function?

return

put

Incorrect

send

Incorrect

pass

Incorrect

The keyword pass in Python is used for a different purpose.

Returning values from functionsReplay

Review this video

Returning values from functions

4m 45s

Question 11 of 15

What will the following program print?

def mult\_inv(num):

return 1/num

result=mult\_inv(3)

print(result)

result

Incorrect

1/num

Incorrect

3

Incorrect

Since the function returns 1 divided by the provided number, it will not return the number itself unless the number is also 1.

0.333333333

Returning values from functionsReplay

Review this video

Returning values from functions

4m 45s

Question 12 of 15

What is the output of the following program?

def isEven(num):

return num % 2 == 0

if isEven(3):

print("3 is even")

else:

print("3 is not even")

3 is even

Incorrect

3 is not even

Correct

There is no output.

Question 13 of 15

Given the first line of the function definition below, which programming language is this?

def factorial(number)

Ruby

Java

Incorrect

C++

Incorrect

Python

Incorrect

This cannot be the first line of a Python function definition, since it does not end with a semicolon.

Functions across languagesReplay

Review this video

Functions across languages

3m 26s

Question 14 of 15

Which value is not an argument to the "hello" function?

def hello(name):

print("Hey,", name)

hello("John")

hello("Mya")

Hey

Correct

Mya

John

Question 15 of 15

In this function definition, the variable "name" is \_\_\_\_\_.

def hello(name):

print("Hey,", name)

a parameter

a function body

Incorrect

an argument

Incorrect

## 6. Conclusion

### 1. Exploring languages

### 2. More learning possibilities

## More Learning Possibilities



Congratulations! Now that you’ve had a first detailed exploration of programming, you’re ready to dig deeper.

The most likely next step is [Programming Foundations: Beyond the Fundamentals](https://www.linkedin.com/learning/programming-foundations-beyond-the-fundamentals?u=83641554), which was designed explicitly to build on this course. If you’d like to check out the basics of programming more broadly, you may want to explore the [Become a Programmer: Foundations](https://www.linkedin.com/learning/paths/become-a-programmer-foundations?u=83641554) learning path, which starts with this course.

Or, if you’re intrigued specifically by the possibilities of Python, you may want to jump to [Learning Python](https://www.linkedin.com/learning/learning-python-14393370/learning-python-22821893?u=104) or [Python for Non-Programmers](https://www.linkedin.com/learning/python-for-non-programmers?u=83641554).