# 2026 100% Free PCEP-30-02–100% Free Certificate Exam | PCEP - Certified Entry-Level Python Programmer Valid Test Answers
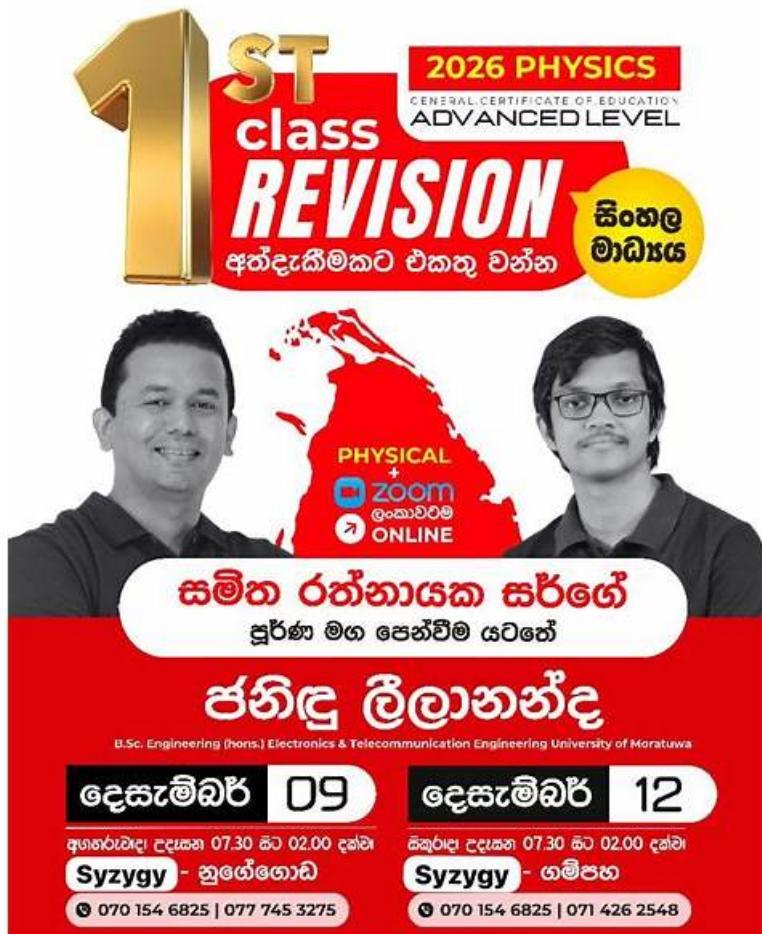
We provide 24-hour online service for all customers who have purchased PCEP-30-02 test guide. You can send us an email to ask questions at anytime, anywhere. For any questions you may have during the use of PCEP-30-02 exam questions, our customer service staff will be patient to help you to solve them. At the same time, if you have problems with downloading and installing, PCEP-30-02 Torrent prep also has dedicated staff that can provide you with remote online guidance. In order to allow you to use our products with confidence, PCEP-30-02 test guide provide you with a 100% pass rate guarantee. Once you unfortunately fail the exam, we will give you a full refund, and our refund process is very simple.

## Python Institute PCEP-30-02 Exam Syllabus Topics:

| Topic | Details |
|-------|---------|
| Topic 1 | • Loops: while, for, range(), loops control, and nesting of loops. |
| Topic 2 | • parameters, arguments, and scopes. It also covers Recursion, Exception hierarchy, Exception handling, etc. |
| Topic 3 | • Functions and Exceptions: This part of the exam covers the definition of function and invocation |

# PCEP - Certified Entry-Level Python Programmer brain dumps, PCEP-30-02 dumps pdf

Are you still worried about not able to pass PCEP-30-02 exam certification? Then you can ask PassTestking for help. It can bring you the master of the sophisticated techniques of IT industry and help you pass PCEP-30-02 certification exam easily. With PassTestking's efforts for years, the passing rate of PCEP-30-02 Certification Exam has reached as high as 100%. Choosing PassTestking is to choose the way to go to a beautiful future.

# Python Institute PCEP - Certified Entry-Level Python Programmer Sample Questions (Q38-Q43):

**NEW QUESTION # 38**
Drag and drop the literals to match their data type names.
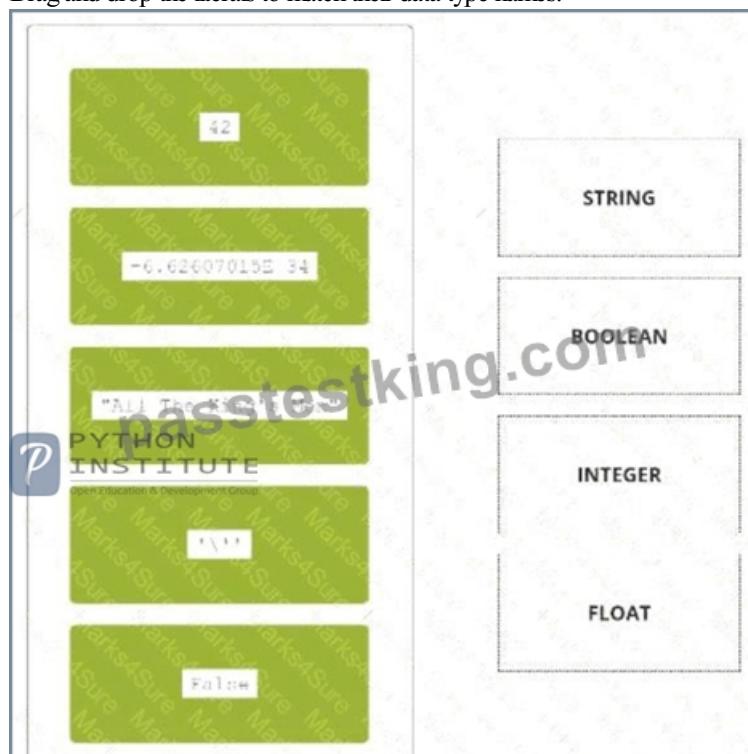


**Answer:**

Explanation:
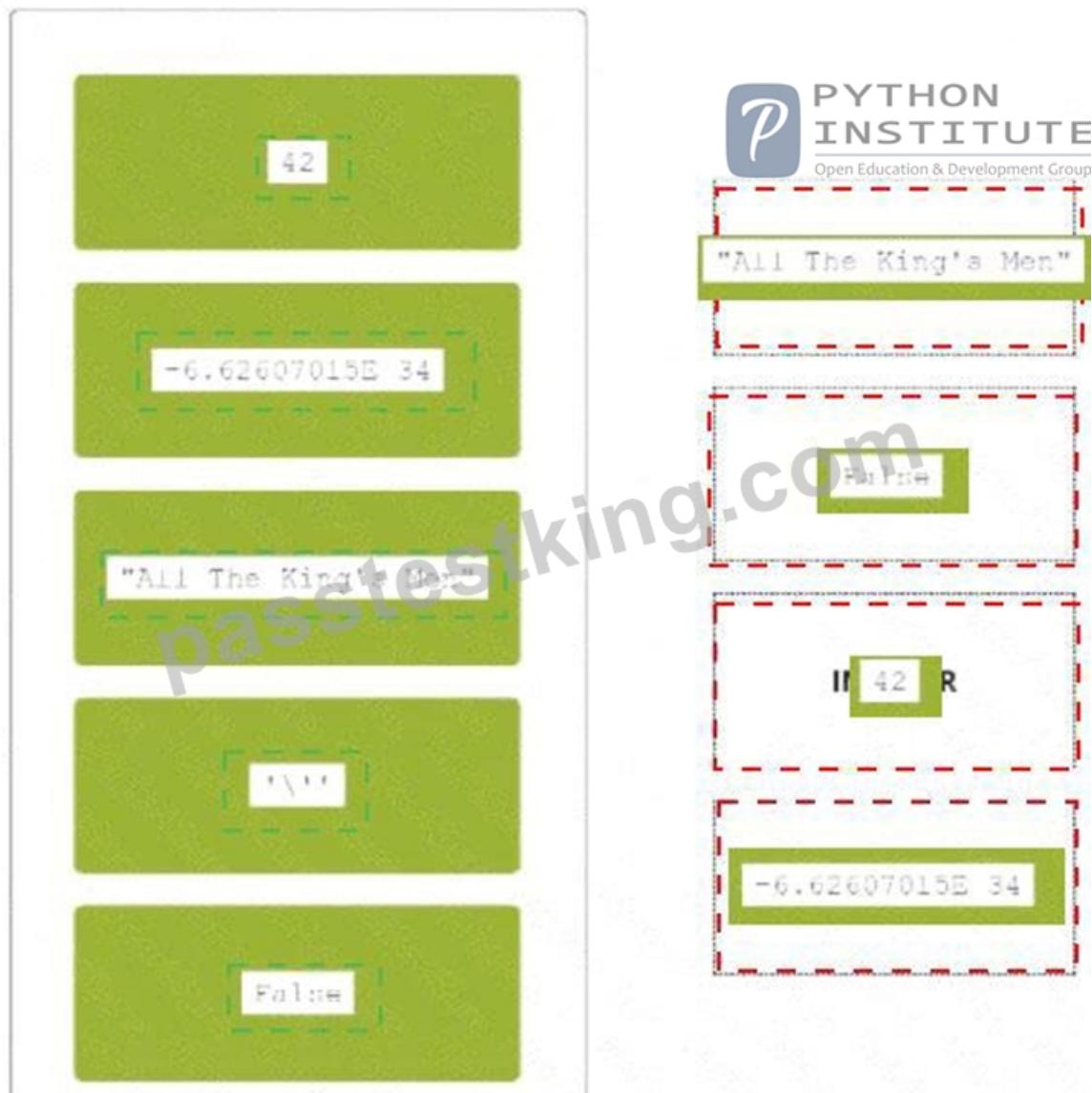
Explanation:

One possible way to drag and drop the literals to match their data type names is:

* STRING: "All The King's Men"
* BOOLEAN: False
* INTEGER: 42
* FLOAT: -6.62607015E-34

A literal is a value that is written exactly as it is meant to be interpreted by the Python interpreter. A data type is a category of values that share some common characteristics or operations. Python has four basic data types: string, boolean, integer, and float.

A string is a sequence of characters enclosed by either single or double quotes. A string can represent text, symbols, or any other information that can be displayed as text. For example, "All The King's Men" is a string literal that represents the title of a novel.

A boolean is a logical value that can be either True or False. A boolean can represent the result of a comparison, a condition, or a logical operation. For example, False is a boolean literal that represents the opposite of True.

An integer is a whole number that can be positive, negative, or zero. An integer can represent a count, an index, or any other quantity that does not require fractions or decimals. For example, 42 is an integer literal that represents the answer to life, the universe, and everything.

A float is a number that can have a fractional part after the decimal point. A float can represent a measurement, a ratio, or any other quantity that requires precision or approximation. For example,
-6.62607015E-34 is a float literal that represents the Planck constant in scientific notation.

You can find more information about the literals and data types in Python in the following references:

* [Python Data Types]
* [Python Literals]
* [Python Basic Syntax]

**NEW QUESTION # 39**
Which of the following functions can be invoked with two arguments?

- A. 

- B. 

- C. 

- D. 

**Answer: C**

Explanation:
The code snippets that you have sent are defining four different functions in Python. A function is a block of code that performs a specific task and can be reused in the program. A function can take zero or more arguments, which are values that are passed to the function when it is called. A function can also return a value or None, which is the default return value in Python.
To define a function in Python, you use the def keyword, followed by the name of the function and parentheses. Inside the parentheses, you can specify the names of the parameters that the function will accept.
After the parentheses, you use a colon and then indent the code block that contains the statements of the function. For example:
def function_name(parameter1, parameter2): # statements of the function return value To call a function in Python, you use the name of the function followed by parentheses. Inside the parentheses, you can pass the values for the arguments that the function expects.
The number and order of the arguments must match the number and order of the parameters in the function definition, unless you use keyword arguments or default values. For example:
function_name(argument1, argument2)
The code snippets that you have sent are as follows:
A) def my_function(): print("Hello")
B) def my_function(a, b): return a + b
C) def my_function(a, b, c): return a * b * c
D) def my_function(a, b=0): return a - b
The question is asking which of these functions can be invoked with two arguments. This means that the function must have two parameters in its definition, or one parameter with a default value and one without.
The default value is a value that is assigned to a parameter if no argument is given for it when the function is called. For example, in option D, the parameter b has a default value of 0, so the function can be called with one or two arguments.
The only option that meets this criterion is option B. The function in option B has two parameters, a and b, and returns the sum of them. This function can be invoked with two arguments, such as my_function(2, 3), which will return 5.
The other options cannot be invoked with two arguments. Option A has no parameters, so it can only be called with no arguments, such as my_function(), which will print "Hello". Option C has three parameters, a, b, and c, and returns the product of them. This function can only be called with three arguments, such as my_function(2, 3, 4), which will return 24. Option D has one parameter with a default value, b, and one without, a, and returns the difference of them. This function can be called with one or two arguments, such as my_function(2) or my_function(2, 3), which will return 2 or -1, respectively.
Therefore, the correct answer is B. Option B.

**NEW QUESTION # 40**
A set of rules which defines the ways in which words can be coupled in sentences is called:

- A. lexis
- B. dictionary
- C. syntax
- D. semantics

**Answer: C**

Explanation:
Explanation
Syntax is the branch of linguistics that studies the structure and rules of sentences in natural languages. Lexis is the vocabulary of a language. Semantics is the study of meaning in language. A dictionary is a collection of words and their definitions, synonyms, pronunciations, etc.


**NEW QUESTION # 41**
Which of the following functions can be invoked with two arguments?

- A.
  

- B.
  

- C.
  

- D.
  

**Answer: B**

Explanation:
Explanation
The code snippets that you have sent are defining four different functions in Python. A function is a block of code that performs a specific task and can be reused in the program. A function can take zero or more arguments, which are values that are passed to the function when it is called. A function can also return a value or None, which is the default return value in Python.
To define a function in Python, you use the def keyword, followed by the name of the function and parentheses. Inside the parentheses, you can specify the names of the parameters that the function will accept.
After the parentheses, you use a colon and then indent the code block that contains the statements of the function. For example:
def function_name(parameter1, parameter2): # statements of the function return value To call a function in Python, you use the name of the function followed by parentheses. Inside the parentheses, you can pass the values for the arguments that the function expects. The number and order of the arguments must match the number and order of the parameters in the function definition, unless you use keyword arguments or default values. For example:
function_name(argument1, argument2)
The code snippets that you have sent are as follows:
A) def my_function(): print("Hello")
B) def my_function(a, b): return a + b
C) def my_function(a, b, c): return a * b * c
D) def my_function(a, b=0): return a - b
The question is asking which of these functions can be invoked with two arguments. This means that the function must have two parameters in its definition, or one parameter with a default value and one without.
The default value is a value that is assigned to a parameter if no argument is given for it when the function is called. For example, in option D, the parameter b has a default value of 0, so the function can be called with one or two arguments.
The only option that meets this criterion is option B. The function in option B has two parameters, a and b, and returns the sum of them. This function can be invoked with two arguments, such as my_function(2, 3), which will return 5.
The other options cannot be invoked with two arguments. Option A has no parameters, so it can only be called with no arguments, such as my_function(), which will print "Hello". Option C has three parameters, a, b, and c, and returns the product of them. This function can only be called with three arguments, such as my_function(2, 3, 4), which will return 24. Option D has one parameter with a default value, b, and one without, a, and returns the difference of them. This function can be called with one or two arguments,

such as my_function(2) or my_function(2, 3), which will return 2 or -1, respectively.

Therefore, the correct answer is B. Option B.

**NEW QUESTION # 42**

What happens when the user runs the following code?

```
speed = 3
while speed < 8:
    speed += 2
    if speed == 7:
        continue
    print("*", end="")
else:
    print("*")
```

- A. The program outputs five asterisks ( ***** ) to the screen.
- B. The program enters an infinite loop.
- C. The program outputs three asterisks( *** ) to the screen.
- D. The program outputs one asterisk ( * ) to the screen.

**Answer: D**

**NEW QUESTION # 43**

......

Originating the PCEP-30-02 exam questions of our company from tenets of offering the most reliable backup for customers, and outstanding results have captured exam candidates' heart for their functions. Our PCEP-30-02 practice materials can be subdivided into three versions. All those versions of usage has been well-accepted by them. There is not much disparity among these versions of PCEP-30-02 simulating practice, but they do helpful to beef up your capacity and speed up you review process to master more knowledge about the exam, so the review process will be unencumbered.

**PCEP-30-02 Valid Test Answers**: https://www.passtestking.com/Python-Institute/PCEP-30-02-practice-exam-dumps.html

- ▢ Easily obtain free download of ➡ PCEP-30-02 ▢▢▢ by searching on ➵ www.pdfvce.com ▢ ▢New PCEP-30-02 Exam Question
- PCEP-30-02 Valid Test Topics ▢ Exam PCEP-30-02 Study Guide ▢ Reliable PCEP-30-02 Exam Syllabus ▢ Enter ▢ www.troytecdumps.com ▢ and search for ✔ PCEP-30-02 ▢✔▢ to download for free ▢New PCEP-30-02 Exam Question
- myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, tutorialbangla.com, academy.pestshop.ng, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, study.stcs.edu.np, rocourses.in, Disposable vapes

BTW, DOWNLOAD part of PassTestking PCEP-30-02 dumps from Cloud Storage: https://drive.google.com/open?id=1AsVyulXm7vRR62PIL01nvVxw8nD_aF5X