



PYTHON INSTITUTE PCEP

**Python Institute Entry-Level Python Programmer Certification
Questions & Answers**

Exam Summary – Syllabus – Questions

PCEP

[Python Institute Certified Entry-Level Python Programmer \(PCEP\)](#)

30 Questions Exam – 70% Cut Score – Duration of 40 minutes

Table of Contents:

Know Your PCEP Certification Well:	2
Python Institute PCEP Entry-Level Python Programmer Certification Details:	2
PCEP Syllabus:	3
Computer Programming and Python Fundamentals (18%)	3
Control Flow – Conditional Blocks and Loops (29%)	4
Data Collections – Tuples, Dictionaries, Lists, and Strings (25%)	4
Functions and Exceptions (28%)	5
Python Institute PCEP Sample Questions:.....	6
Study Guide to Crack Python Institute Entry-Level Python Programmer PCEP Exam:.....	9

Know Your PCEP Certification Well:

The PCEP is best suitable for candidates who want to gain knowledge in the Python Institute Entry-Level Programmer. Before you start your PCEP preparation you may struggle to get all the crucial Entry-Level Python Programmer materials like PCEP syllabus, sample questions, study guide.

But don't worry the PCEP PDF is here to help you prepare in a stress free manner. The PDF is a combination of all your queries like-

- What is in the PCEP syllabus?
- How many questions are there in the PCEP exam?
- Which Practice test would help me to pass the PCEP exam at the first attempt?

Passing the PCEP exam makes you Python Institute Certified Entry-Level Python Programmer. Having the Entry-Level Python Programmer certification opens multiple opportunities for you. You can grab a new job, get a higher salary or simply get recognition within your current organization.

Python Institute PCEP Entry-Level Python Programmer Certification Details:

Exam Name	Python Institute Certified Entry-Level Python Programmer (PCEP)
Exam Code	PCEP
Exam Price	\$59 (USD)
Duration	40 mins
Number of Questions	30
Passing Score	70%
Books / Training	Python Essentials 1 PCAP Programming Essentials in Python
Schedule Exam	Pearson VUE
Sample Questions	Python Institute Entry-Level Python Programmer Sample Questions
Practice Exam	Python Institute PCEP Certification Practice Exam

PCEP Syllabus:

Topic	Details
Computer Programming and Python Fundamentals (18%)	
Understand fundamental terms and definitions	<ul style="list-style-type: none"> - interpreting and the interpreter, compilation and the compiler - lexis, syntax, and semantics
Understand Python's logic and structure	<ul style="list-style-type: none"> - keywords - instructions - indentation - comments
Introduce literals and variables into code and use different numeral systems	<ul style="list-style-type: none"> - Boolean, integers, floating-point numbers - scientific notation - strings - binary, octal, decimal, and hexadecimal numeral systems - variables - naming conventions - implementing PEP-8 recommendations
Choose operators and data types adequate to the problem	<ul style="list-style-type: none"> - numeric operators: <code>** * / % // + -</code> - string operators: <code>* +</code> - assignment and shortcut operators - unary and binary operators - priorities and binding - bitwise operators: <code>~ & ^ << >></code> - Boolean operators: <code>not, and, or</code> - Boolean expressions - relational operators (<code>== != > >= < <=</code>) - the accuracy of floating-point numbers - type casting
Perform Input/Output console operations	<ul style="list-style-type: none"> - the <code>print()</code> and <code>input()</code> functions - the <code>sep=</code> and <code>end=</code> keyword parameters - the <code>int()</code> and <code>float()</code> functions

Topic	Details
Control Flow – Conditional Blocks and Loops (29%)	
Make decisions and branch the flow with the if instruction	<ul style="list-style-type: none"> - conditional statements: if, if-else, if-elif, if-elif-else - multiple conditional statements - nesting conditional statements
Perform different types of iterations	<ul style="list-style-type: none"> - the pass instruction - building loops with while, for, range(), and in - iterating through sequences - expanding loops with while-else and for-else - nesting loops and conditional statements - controlling loop execution with break and continue
Data Collections – Tuples, Dictionaries, Lists, and Strings (25%)	
Collect and process data using lists	<ul style="list-style-type: none"> - constructing vectors - indexing and slicing - the len() function - list methods: append(), insert(), index(), etc. - functions: len(), sorted() - the del instruction - iterating through lists with the for loop - initializing loops - the in and not in operators - list comprehensions - copying and cloning - lists in lists: matrices and cubes
Collect and process data using tuples	<ul style="list-style-type: none"> - tuples: indexing, slicing, building, immutability - tuples vs. lists: similarities and differences - lists inside tuples and tuples inside lists
Collect and process data using dictionaries	<ul style="list-style-type: none"> - dictionaries: building, indexing, adding and removing keys - iterating through dictionaries and their keys and values - checking the existence of keys - methods: keys(), items(), and values()
Operate with strings	<ul style="list-style-type: none"> - constructing strings - indexing, slicing, immutability

Topic	Details
	<ul style="list-style-type: none"> - escaping using the \ character - quotes and apostrophes inside strings - multi-line strings - basic string functions and methods
Functions and Exceptions (28%)	
Decompose the code using functions	<ul style="list-style-type: none"> - defining and invoking user-defined functions and generators - the return keyword, returning results - the None keyword - recursion
Organize interaction between the function and its environment	<ul style="list-style-type: none"> - parameters vs. arguments - positional, keyword, and mixed argument passing - default parameter values - name scopes, name hiding (shadowing), and the global keyword
Python Built-In Exceptions Hierarchy	<ul style="list-style-type: none"> - BaseException - Exception - SystemExit - KeyboardInterrupt - abstract exceptions - ArithmeticError - LookupError - IndexError - KeyError - TypeError - ValueError
Basics of Python Exception Handling	<ul style="list-style-type: none"> - try-except / the try-except Exception - ordering the except branches - propagating exceptions through function boundaries - delegating responsibility for handling exceptions

Python Institute PCEP Sample Questions:

Question: 1

An integer number preceded by an 0x (Zero-x) will be treated as:

- a) Octal
- b) Binary
- c) Hexadecimal
- d) Decimal

Answer: c

Question: 2

Which of the following are correct statements?

- a) True + 1 evaluates to 2
- b) True and False evaluates to False
- c) True or False evaluates to False
- d) 7+ False evaluates to False

Answer: a, b

Question: 3

Who created Python?

- a) Guido ban Rossum
- b) Guido van Rossum
- c) Guido the Russian
- d) Guodo van Rossum

Answer: b

Question: 4

Python name comes from which of the following?

- a) Python Café
- b) Python Forest
- c) Python snake
- d) Monty Python's Flying Circus

Answer: d

Question: 5

If a list passed into function's argument and modified inside the function:

- a) Will affect the argument
- b) Will not affect the argument
- c) Will give an error
- d) Will become global by default

Answer: a

Question: 6

A complete set of commands is known as:

- a) Instruction list
- b) Code laws
- c) Command-line
- d) Command list

Answer: a

Question: 7

The meaning of positional parameter is determined by:

- a) Position
- b) Name
- c) Style
- d) None

Answer: a

Question: 8

Which of the following is incorrect for a dictionary in Python?

- a) each key must be unique
- b) the key should be an immutable object
- c) the len() function returns the sum of key-value elements in the dictionary
- d) the len() function returns the numbers of key-value elements in the dictionary

Answer: c

Question: 9

What does the method items() returns in Python Dictionary?

- a) The method items() returns the lists
- b) The method items() returns the tuples
- c) The method items() returns the keys in a list
- d) The method items() returns the values in a list

Answer: b

Question: 10

Octal has the following base:

- a) 2
- b) 8
- c) 10
- d) 16

Answer: b

Study Guide to Crack Python Institute Entry-Level Python Programmer PCEP Exam:

- Getting details of the PCEP syllabus, is the first step of a study plan. This pdf is going to be of ultimate help. Completion of the syllabus is must to pass the PCEP exam.
- Making a schedule is vital. A structured method of preparation leads to success. A candidate must plan his schedule and follow it rigorously to attain success.
- Joining the Python Institute provided training for PCEP exam could be of much help. If there is specific training for the exam, you can discover it from the link above.
- Read from the PCEP sample questions to gain your idea about the actual exam questions. In this PDF useful sample questions are provided to make your exam preparation easy.
- Practicing on PCEP practice tests is must. Continuous practice will make you an expert in all syllabus areas.

Reliable Online Practice Test for PCEP Certification

Make EduSum.com your best friend during your Python Institute Certified Entry-Level Python Programmer exam preparation. We provide authentic practice tests for the PCEP exam. Experts design these online practice tests, so we can offer you an exclusive experience of taking the actual PCEP exam. We guarantee you 100% success in your first exam attempt if you continue practicing regularly. Don't bother if you don't get 100% marks in initial practice exam attempts. Just utilize the result section to know your strengths and weaknesses and prepare according to that until you get 100% with our practice tests. Our evaluation makes you confident, and you can score high in the PCEP exam.

Start Online Practice of PCEP Exam by visiting URL

<https://www.edusum.com/python-institute/pcep-python-institute-certified-entry-level-python-programmer>