

Quiz WGU - Updated Foundations-of-Computer-Science - Latest WGU Foundations of Computer Science Exam Forum

**WGU FOUNDATIONS OF COMPUTER
SCIENCE EXAM WITH CORRECT ACTUAL
QUESTIONS AND CORRECTLY WELL
DEFINED ANSWERS LATEST ALREADY
GRADED A+ 2025 - 2026**

What are the two most common states when assigning permissions to a particular network location?

Allow and deny

Which Python code will delete the first value in a list (list name 'list')?

del list[0]

What feature causes a device driver to be a 'signed' driver?

It is equipped with a digital signature.

How many different data types are allowed in a given NumPy array?

One

Which file system is commonly used in Windows and supports file permissions?

NTFS

P.S. Free 2026 WGU Foundations-of-Computer-Science dumps are available on Google Drive shared by BraindumpsPass:
https://drive.google.com/open?id=1u7CUniTBm0AAsqLr5YISJyNcSZD-p7_N

Our customers receive WGU Foundations-of-Computer-Science questions updates for up to 365 days after their purchase. They can also try a free demo for satisfaction before buying our WGU Foundations-of-Computer-Science dumps. And a 24/7 support system assists them whenever they are stuck in any problem or issue. This WGU Foundations-of-Computer-Science Questions is a complete package and a blessing for candidates who want to prepare quickly for the Foundations-of-Computer-Science exam. Buy It Now!

WGU Foundations-of-Computer-Science Exam Syllabus Topics:

Section	Objectives
Topic 1: OS Fundamentals	- Demonstrate various techniques and tools to manage operating systems - Describe fundamental principles and core concepts of operating systems - Identify common privacy and security concepts that could be implemented in operating systems
Topic 2: Basic Program Design	- Identify variables and data types within a programming language - Use functions, methods, and packages to leverage programming language - Explain how to store, access, and manipulate data in lists

Topic 3: Algorithm Efficiency	- Describe the relationships between algorithm complexity and data structures - Choose an appropriate algorithm searching method based on a given scenario - Choose an appropriate sorting algorithm method based on a given scenario
Topic 4: Data Profiling	- Apply fundamental concepts and subsetting techniques to a dataset - Utilize a programming language to manipulate arrays and discover insights

>> Latest Foundations-of-Computer-Science Exam Forum <<

Foundations-of-Computer-Science Exam Questions - Successful Guidelines For Preparation [2026]

The BraindumpsPass is one of the most in-demand platforms for WGU Foundations-of-Computer-Science exam preparation and success. The BraindumpsPass is offering valid, and real WGU Foundations-of-Computer-Science exam dumps. They all used the WGU Foundations-of-Computer-Science exam dumps and passed their dream WGU Foundations-of-Computer-Science Exam easily. The WGU Foundations-of-Computer-Science exam dumps will provide you with everything that you need to prepare, learn and pass the difficult WGU Foundations-of-Computer-Science exam.

WGU Foundations of Computer Science Sample Questions (Q18-Q23):

NEW QUESTION # 18

What is the likely cause if a default Python configuration does not recognize a NumPy array as an allowed data structure?

- A. The array module is not imported.
- **B. The NumPy package is not present.**
- C. The Python version is outdated.
- D. The Python interpreter is misconfigured.

Answer: B

Explanation:

NumPy arrays are not a built-in Python data structure. In a default Python installation, the interpreter includes core types such as int, float, str, list, tuple, dict, and set, plus the standard library. A NumPy array, typically created as `numpy.ndarray`, is provided by the third-party NumPy library. Therefore, if a "default Python configuration" does not recognize a NumPy array, the most likely cause is that NumPy is not installed or not available in the active environment. This happens often when a user has multiple Python environments (system Python, virtual environments, conda environments) and installs NumPy into one environment while running code in another.

Option B is incorrect because Python's standard-library array module is different from NumPy. Importing array does not create or enable NumPy's ndarray type. Option C is possible in rare cases, but the typical, textbook-aligned explanation is missing dependencies rather than an incorrectly configured interpreter. Option D is also unlikely: while very old Python versions may cause compatibility issues with modern NumPy releases, the symptom described-NumPy arrays not being recognized at all-more directly indicates the package is absent in the running environment.

In practice, verifying `import numpy` and checking the installed packages for the current interpreter resolves the issue.

NEW QUESTION # 19

What is the expected result of running the following code: `list1[0] = "California"`?

- A. The list will be extended by adding "California" at the end.
- **B. The first value in the list will be replaced with "California".**
- C. A second element will be added to the line "California".
- D. A new list will be created with the value "California".

Answer: B

Explanation:

Python lists are mutable sequences, which means elements can be changed in place after the list has been created. The expression `list1[0] = "California"` uses indexing to target the element at position 0 (the first element, because Python uses zero-based indexing) and assignment (=) to replace that element with a new value. As a result, the list keeps the same length, but its first entry becomes "California".

This operation does not create a new list (so option A is incorrect); it modifies the existing list object referenced by list1. It also does not append to the end of the list (so option C is incorrect). Appending would use methods like `list1.append("California")`. Option D is not meaningful in Python list semantics; assignment to a single index replaces exactly one element rather than "adding a second element to the line." Textbooks highlight this difference between mutable and immutable sequence types. For example, strings are immutable, so you cannot assign to `some_string[0]`. Lists, however, are designed for collections that change over time, supporting updates, insertions, deletions, and reordering. Index assignment is fundamental for many algorithms: updating an array-like buffer, modifying a dataset row, replacing incorrect values, or implementing in-place transformations efficiently.

NEW QUESTION # 20

The `np_2d` array stores information about multiple family members. Each row represents a different person, and the columns store family member attributes in the following order:

Age (years)

Weight (pounds)

Height (inches)

How is the weight of all family members selected from the `np_2d` array?

- A. `np_2d[:, 1]`
- B. `np_2d[2, :]`
- C. `np_2d[:, 2]`
- D. `np_2d[1, :]`

Answer: A

Explanation:

In a 2D NumPy array, rows and columns represent different dimensions of the data. The indexing form `array[row_selection, column_selection]` allows you to select entire rows, entire columns, or submatrices. The slice `:` means "all indices along this dimension." Since each row corresponds to a family member (a person), selecting weights for all family members means selecting all rows for the weight column.

The problem states the columns are ordered as: Age (column 0), Weight (column 1), Height (column 2).

Therefore, the weight column has index 1. The expression `np_2d[:, 1]` uses `:` to take every row and 1 to take the second column, producing a 1D array (or a column view) containing the weight values for all people.

Option A, `np_2d[:, 2]`, would select the height column, not weight. Option C, `np_2d[2, :]`, selects the third row (the third person) and all columns—age, weight, and height—for just that one person. Option D, `np_2d[1, :]`, selects the second person's entire row.

This column selection technique is fundamental in data analysis because datasets are often stored as

"rows = observations, columns = features," and extracting a feature vector is a frequent operation before computing statistics or building models.

NEW QUESTION # 21

What Python code would return the value 40 from `np_2d`, where `np_2d = np.array([[1, 2, 3, 4], [10, 20, 30, 40]])`?

- A. `np_2d[1, 3]`
- B. `np_2d[3, 1]`
- C. `np_2d[4, 1]`
- D. `np_2d[0, 4]`

Answer: A

Explanation:

In a 2D NumPy array, indexing is written as `array[row_index, column_index]` using zero-based indices. The array `np_2d = np.array([[1, 2, 3, 4], [10, 20, 30, 40]])` has two rows (indices 0 and 1) and four columns (indices 0, 1, 2, 3). The value 40 is located in the second row and the fourth column. Using zero-based indexing, that corresponds to row index 1 and column index 3. Therefore, `np_2d[1, 3]` returns 40.

Option A attempts to access row 3, which does not exist and would raise an `IndexError`. Option C attempts to access column 4 in row 0, but valid column indices are only 0 through 3, so it would also error. Option D likewise refers to a non-existent row 4. Only option B uses valid indices and points to the correct location.

Textbooks emphasize multi-dimensional indexing because it underlies matrix operations, dataset manipulation, and feature extraction in data science. Correctly interpreting rows and columns is essential when rows represent observations (like people) and columns represent attributes (like age, weight, height). This question tests precise control over row/column addressing, which prevents subtle

bugs in numerical analysis.

NEW QUESTION # 22

Which Python function is used to display the data type of a given variable?

- A. Data()
- B. Show()
- C. GetVar()
- D. type()

Answer: D

Explanation:

Python is a dynamically typed language, meaning variables do not require explicit type declarations; instead, objects carry type information at runtime. To inspect the type of an object, Python provides the built-in function `type()`. When you pass a variable or value into `type()`, it returns the object's class, which represents its data type. For example, `type(5)` returns `<class 'int'>`, `type(3.14)` returns `<class 'float'>`, and `type("hello")` returns `<class 'str'>`. This is commonly used in debugging, learning exercises, and when writing functions that must behave differently depending on input types.

Textbook discussions often pair `type()` with Python's object model: everything in Python is an object, and each object is an instance of some class. `type()` reveals that class. In addition, `type()` can be used in more advanced ways, such as dynamic class creation, but its foundational educational use is type inspection.

The other options are not correct because `GetVar()`, `Show()`, and `Data()` are not standard Python built-ins for type checking. While developers can define functions with those names, they are not part of Python's core language or standard library in the sense required by the question. For typical coursework and professional Python usage, the correct and universally accepted function is `type()`.

NEW QUESTION # 23

.....

If you feel that you always suffer from procrastination and cannot make full use of your spare time, maybe our Foundations-of-Computer-Science study materials can help you solve your problem. We are willing to recommend you to try the Foundations-of-Computer-Science study materials from our company. Our Foundations-of-Computer-Science training guide are high quality and efficiency test tools for all people. If you buy our Foundations-of-Computer-Science Preparation questions, we can promise that you can use our Foundations-of-Computer-Science study materials for study in anytime and anywhere. Because we have three version of Foundations-of-Computer-Science exam questions that can satisfy all needs of our customers.

Foundations-of-Computer-Science Printable PDF: <https://www.braindumps.com/WGU/Foundations-of-Computer-Science-practice-exam-dumps.html>

- Foundations-of-Computer-Science Exam Certification Cost ☐ Pass4sure Foundations-of-Computer-Science Pass Guide ☐ Foundations-of-Computer-Science Exam Certification Cost ☐ Copy URL ☐ www.examcollectionpass.com ☐ open and search for 《 Foundations-of-Computer-Science 》 to download for free ☐ Exam Sample Foundations-of-Computer-Science Online
- Foundations-of-Computer-Science Authentic Exam Hub ☐ Updated Foundations-of-Computer-Science Testkings ☐ Pass4sure Foundations-of-Computer-Science Pass Guide ↵ Enter ▷ www.pdfvce.com ◁ and search for ☐ Foundations-of-Computer-Science ☐ to download for free ☐ Pass4sure Foundations-of-Computer-Science Pass Guide
- WGU Commitment to Your Foundations-of-Computer-Science WGU Foundations of Computer Science Exam Success ☐ Search for ► Foundations-of-Computer-Science ◀ on ➡ www.pdfdumps.com ☐ immediately to obtain a free download ☐ Latest Foundations-of-Computer-Science Exam Camp
- Well-Prepared Latest Foundations-of-Computer-Science Exam Forum - Effective Foundations-of-Computer-Science Exam Tool Guarantee Purchasing Safety ↔ Search for ☐ Foundations-of-Computer-Science ☐ and obtain a free download on **【 www.pdfvce.com 】** ☐ Foundations-of-Computer-Science Exam Guide Materials
- Up-to-Date Online WGU Foundations-of-Computer-Science Practice Test Engine ☐ Enter ☀ www.prep4away.com ☐ ☀ ☐ and search for ➡ Foundations-of-Computer-Science ☐ ☐ to download for free ☐ Latest Foundations-of-Computer-Science Exam Camp
- Pass4sure Foundations-of-Computer-Science Pass Guide ☐ Latest Foundations-of-Computer-Science Exam Camp ☐ Foundations-of-Computer-Science Latest Braindumps Pdf ☐ Open ✓ www.pdfvce.com ☐ ✓ ☐ and search for ☐ Foundations-of-Computer-Science ☐ to download exam materials for free ☐ Exam Sample Foundations-of-Computer-Science Online

- BTW, DOWNLOAD part of BraindumpsPass Foundations-of-Computer-Science dumps from Cloud Storage:
https://drive.google.com/open?id=1u7CUiTBm0AAsqLr5YISJyNcSZD-p7_N

BTW, DOWNLOAD part of BraindumpsPass Foundations-of-Computer-Science dumps from Cloud Storage:
https://drive.google.com/open?id=1u7CUiTBm0AAsqLr5YISJyNcSZD-p7_N