PYTHON

Assignment Questions

SKILLS

T

Assignment Questions

1. Who developed Python Programming Language?Python was developed by Guido van Rossum, a Dutch computer programmer, in the late 1980s and early 1990s.

2. Which type of Programming does Python support?

Python supports multiple programming paradigms, including object-oriented programming, procedural programming, and functional programming.

3. Is Python case sensitive when dealing with identifiers?

Yes, Python is case sensitive when dealing with identifiers. For example, a variable named "myVariable" is different from a variable named "MyVariable".

4. What is the correct extension of the Python file?

The correct extension for Python files is .py. For example, a Python file named "myfile" would have the following file name: "myfile.py".

5. Is Python code compiled or interpreted?

Python code is interpreted, not compiled. This means that the code is executed line by line at runtime by the interpreter, rather than being transformed into machine code before being executed by the computer.

6. Name a few blocks of code used to define in Python language?

In Python, code blocks are defined using indentation. Some common code blocks include:

1. Functions: defined using the "def" keyword, followed by the function name and its parameters.
2. Classes: defined using the "class" keyword, followed by the class name and its methods.
3. Loops: including "for" loops and "while" loops, which are used to repeatedly execute a block of code until a certain condition is met.
4. Conditionals: including "if" statements and "if-else" statements, which are used to

conditionally execute a block of code based on a specific condition.

7. State a character used to give single-line comments in Python?

The pound sign or hash symbol (#) is used to give single-line comments in Python. Everything after the # on a line is considered a comment and ignored by the interpreter.

For example # This is a single-line comment in Python

8. Mention functions which can help us to find the version of python that we are currently working on?

There are two built-in functions in Python that can be used to find the version of Python that you are currently working on:

1. sys.version: returns a string that contains information about the version of Python, including the version number, build date, and compiler used.
2. sys.version_info: returns a named tuple that contains information about the version of Python, including the major, minor, and micro version numbers, as well as any release level flags.

For example:

```
import sys

print("Using Python version:", sys.version)
print("Version info:", sys.version_info)
```

9. Python supports the creation of anonymous functions at runtime, using a construct called

"lambda." The lambda construct in Python allows you to create anonymous functions, or functions without a name, at runtime. Lambda functions can be used wherever function objects are required and are often used in functional programming constructs, such as map, filter, and reduce.

10. What does pip stand for python?

"pip" stands for "Pip Installs Packages."

11. Mention a few built-in functions in python?

Python has many built-in functions, some of the most commonly used ones are:

1. print(): used to display output to the console.
2. len(): used to get the length of a data structure, such as a string, list, or tuple.
3. range(): used to generate a sequence of numbers.
4. int() and float(): used to convert data from one data type to another, such as from a string to an integer or from a float to an integer.
5. str(): used to convert data to a string data type.
6. sum(): used to calculate the sum of a sequence of numbers.
7. max() and min(): used to find the maximum and minimum values in a data structure, respectively.
8. sorted(): used to sort a data structure, such as a list.
9. type(): used to determine the data type of an object.
10. abs(): used to calculate the absolute value of a number.

These are just a few examples, there are many other built-in functions available in Python, each with its own specific use case.

12. What is the maximum possible length of an identifier in Python?

There is no specific limit to the length of an identifier in Python. An identifier can be any combination of letters, numbers, and underscores, and can be as long as memory allows. However, it is good practice to keep identifier names descriptive and concise.

13. What are the benefits of using Python?

There are many benefits to using Python, including:

1. Easy to Learn and Read: Python has a simple and straightforward syntax, making it easy to learn for both beginners and experienced programmers.
2. Versatile: Python can be used for a wide range of tasks, including web

development, scientific computing, data analysis, artificial intelligence, and more.
3. Large Community and Support: Python has a large and active community of developers, which provides a wealth of resources and support.
4. Many Libraries and Tools: Python has a large number of libraries and tools available, which makes it easier to perform complex tasks and extend the functionality of the language.
5. Dynamic Typing: Python is dynamically typed, meaning that you don't have to declare variable data types before using them. This can make the coding process faster and more flexible.
6. High-level: Python provides high-level data structures and easy-to-use syntax, making it easier to write code that is more concise and less prone to errors.
7. Cross-platform: Python runs on many different platforms, including Windows, macOS, Linux, and more, making it a great choice for cross-platform development.

These are just a few of the benefits of using Python. Overall, Python is a powerful and flexible language that is well-suited for a wide range of tasks, making it a popular choice for both personal and professional projects.

14. How is memory managed in Python?

Memory management in Python involves the management of a private heap. A private heap is a portion of memory that is exclusive to the Python process. All Python objects and data structures are stored in the private heap. The operating system cannot allocate this piece of memory to another process.

15. How to install Python on Windows and set path variables? 16. Is indentation required in python?

1. Installing Python on Windows:
- Go to the official Python website (https://www.python.org/downloads/) and download the latest version of Python for Windows.
- Run the downloaded .exe file and follow the installation steps.
- Once installation is complete, open the Command Prompt and type "python" to check if Python is installed and working correctly.
2. Setting Path Variables:
- Open the Start menu and search for "Environment Variables".
- Click on "Edit the system environment variables".
- Click on the "Environment Variables" button.
- Under "System Variables", scroll down and find the "Path" variable, then click on "Edit".
- Click on "New" and add the path to the Python installation (e.g., "C:\Python37").

- Click "OK" to save the changes.

3.Indentation in Python:

- Yes, indentation is required in Python.
- Indentation is used to define blocks of code and is an important part of Python's syntax.

P

W

SKILLS

Data Science Masters