Important Viva voce Questions of Artificial Intelligence with Python

1. **What do you understand by Artificial Intelligence?**

   Artificial intelligence is computer science technology that emphasizes creating intelligent machine that can mimic human behaviour. Here Intelligent machines can be defined as the machine that can behave like a human, think like a human, and also capable of decision making. It is made up of two words, "Artificial" and "Intelligence," which means the "man-made thinking ability."

2. **Give some real-world applications of AI.**

   There are various real-world applications of AI, and some of them are given below:

   - Google Search Engine
   - Ridesharing Applications
   - Spam Filters in Email
   - Social Networking
   - Product recommendations

3. **How Artificial intelligence, Machine Learning, and Deep Learning differ from each other?**

The difference between AI, ML, and Deep Learning is given in the below table:

Artificial Intelligence	Machine Learning	Deep Learning
The term Artificial intelligence was first coined in the year **1956 by John McCarthy**.	The term ML was first coined in the year **1959 by Arthur Samuel**.	The term DL was first coined in the year **2000 Igor Aizenberg**.
It is a technology that is used to create intelligent machines that can mimic human behavior.	It is a subset of AI that learns from past data and experiences.	It is the subset of machine learning and AI that is inspired by the human brain cells, called neurons, and imitates the working of the human brain.
AI completely deals with structured, semi-structured data.	ML deals with structured and semi-structured data.	Deep learning deals with structured and unstructured data.
It requires a huge amount of data to work.	It can work with less amount of data compared to deep learning and AI.	It requires a huge amount of the data compared to the ML.
The goal of AI is to enable the machine to think without any human intervention.	The goal of ML is to enable the machine to learn from past experiences.	The goal of deep learning is to solve the complex problems as the human brain does, using various algorithms.

4. **What are the different domains/Subsets of AI?**

AI covers lots of domains or subsets, and some main domains are given below:

- Machine Learning
- Deep Learning
- Neural Network
- Expert System
- Fuzzy Logic
- Natural Language Processing
- Robotics
- Speech Recognition.

5. Which programming language is used for AI?

Below are the top five programming languages that are widely used for the development of Artificial Intelligence:

- Python
- Java
- Lisp
- R
- Prolog

Among the above five languages, Python is the most used language for AI development due to its simplicity and availability of lots of libraries, such as Numpy, Pandas, etc.

6. Which programming language is not generally used in AI, and why?

Perl Programming language is not commonly used language for AI, as it is the scripting language.

7. What is a Chatbot?

A chatbot is Artificial intelligence software or agent that can simulate a conversation with humans or users using Natural language processing. The conversation can be achieved through an application, website, or messaging apps. These chatbots are also called as the digital assistants and can interact with humans in the form of text or through voice.

The AI chatbots are broadly used in most businesses to provide 24*7 virtual customer support to their customers, such as HDFC Eva chatbot, Vainubot, etc.

8. What are the different software platforms for AI development?

- Google Cloud AI platform
- Microsoft Azure AI platform

- IBM Watson
- TensorFlow
- Infosys Nia
- Rainbird
- Dialogflow

Q1. What is Python?
Q2. Python is an interpreted language. Explain

Q3. What is the difference between lists and tuples?

Q4. What is pep 8?

Q5. What are the Key features of Python?

Q6. How is Memory managed in Python?

Q7. What is PYTHONPATH?

Q8. What are Python Modules?

Q9. What are python namespaces?

Q10. Explain Inheritance in Python with an example?

Basic Python Interview Questions for Freshers

1. What is Python?

- Python is a **high-level**, **interpreted**, interactive, and **object-oriented** scripting language. It uses English keywords frequently. Whereas, other languages use punctuation, Python has fewer syntactic constructions.
- Python is designed to be highly **readable** and **compatible** with different platforms such as Mac, Windows, Linux, Raspberry Pi, etc.

2. Python is an interpreted language. Explain.

An interpreted language is any programming language that executes its statements line by line. Programs written in Python run directly from the source code, with no intermediary compilation step.

2. Python is an interpreted language. Explain.

An interpreted language is any programming language that executes its statements line by line. Programs written in Python run directly from the source code, with no intermediary compilation step.

3. What is the difference between lists and tuples?

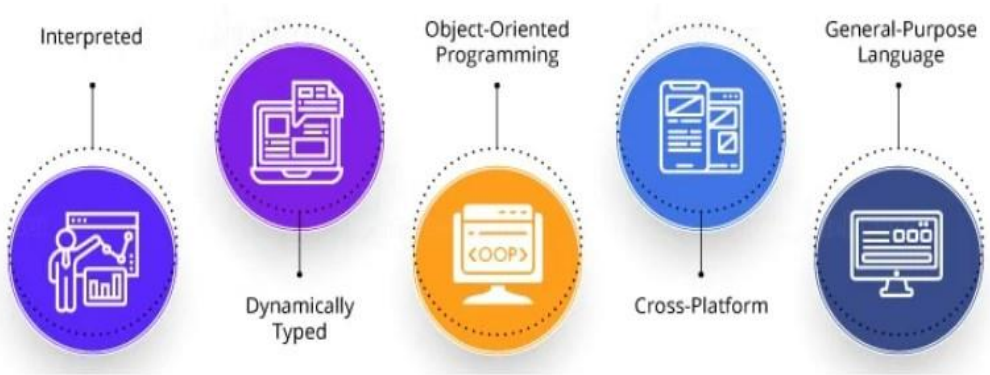
Lists	Tuples
Lists are mutable, i.e., they can be edited	Tuples are immutable (they are lists that cannot be edited)
Lists are usually slower than tuples	Tuples are faster than lists
Lists consume a lot of memory	Tuples consume less memory when compared to lists
Lists are less reliable in terms of errors as unexpected changes are more likely to occur	Tuples are more reliable as it is hard for any unexpected change to occur
Lists consist of many built-in functions.	Tuples do not consist of any built-in functions.
Syntax: list_1 = [10, 'Intellipaat', 20]	Syntax: tup_1 = (10, 'Intellipaat' , 20)

4. What is pep 8?

PEP in Python stands for Python Enhancement Proposal. It is a set of rules that specify how to write and design Python code for maximum readability.

5. What are the Key features of Python?

The key features of Python are as follows:



6. How is Memory managed in Python?

- Memory in Python is managed by Python private heap space. All Python objects and data structures are located in a private heap. This private heap is taken care of by Python Interpreter itself, and a programmer doesn't have access to this private heap.
- Python memory manager takes care of the allocation of Python private heap space.
- Memory for Python private heap space is made available by Python's in-built garbage collector, which recycles and frees up all the unused memory.

7. What is PYTHONPATH?

PYTHONPATH has a role similar to PATH. This variable tells Python Interpreter where to locate the module files imported into a program. It should include the Python source library directory and the directories containing Python source code. PYTHONPATH is sometimes preset by Python Installer.

8. What are Python Modules?

Files containing Python codes are referred to as **Python Modules**. This code can either be classes, functions, or variables and saves the programmer time by providing the predefined functionalities when needed. It is a file with ".py" extension containing an executable code.

Commonly used built modules are listed below:

- os
- sys
- data time
- math
- random
- JSON

9. What are python namespaces?

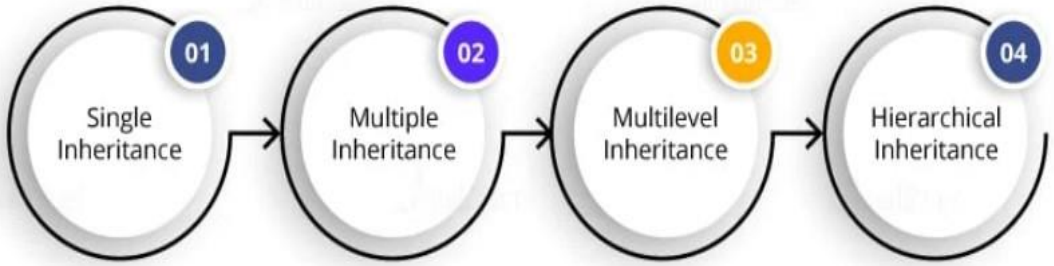
A Python namespace ensures that object names in a program are unique and can be used without any conflict. Python implements these namespaces as dictionaries with 'name as key' mapped to its respective 'object as value'.

Let's explore some examples of namespaces:

- **Local Namespace** consists of local names inside a function. It is temporarily created for a function call and gets cleared once the function returns.
- **Global Namespace** consists of names from various imported modules/packages that are being used in the ongoing project. It is created once the package is imported into the script and survives till the execution of the script.
- **Built-in Namespace** consists of built-in functions of core Python and dedicated built-in names for various types of exceptions.

10. Explain Inheritance in Python with an example?

As Python follows an **object-oriented** programming paradigm, classes in Python have the ability to inherit the properties of another class. This process is known as inheritance. Inheritance provides the **code reusability feature**. The class that is being inherited is called a **superclass** or the parent class, and the class that inherits the superclass is called a **derived** or child class. The following types of inheritance are supported in Python:



- **Single inheritance**: When a class inherits only one superclass
- **Multiple inheritance**: When a class inherits multiple superclasses
- **Multilevel inheritance**: When a class inherits a superclass, and then another class inherits this derived class forming a 'parent, child, and grandchild' class structure
- **Hierarchical inheritance**: When one superclass is inherited by multiple derived classes

60. What do you understand by NumPy?

NumPy is one of the most popular, easy-to-use, versatile, open-source, python-based, general-purpose package that is used for processing arrays. NumPy is short for NUMerical PYthon. This is very famous for its highly optimized tools that result in high performance and powerful N-Dimensional array processing feature that is designed explicitly to work on complex arrays. Due to its popularity and powerful performance and its flexibility to perform various operations like trigonometric operations, algebraic and statistical computations, it is most commonly used in performing scientific computations and various broadcasting functions. The following image shows the applications of NumPy:

Uses Of NumPy