

Python Viva Questions with Answers

1. What are the key features of Python?

- Easy to learn and read
- Interpreted language
- Dynamically typed
- Supports object-oriented and functional programming
- Large standard library

2. What is the difference between 'is' and '=='?

'==' compares values. 'is' compares object identity.

Example:

```
a = [1, 2]
```

```
b = [1, 2]
```

```
a == b  # True
```

```
a is b  # False
```

3. What are Python data types?

- Basic: int, float, str, bool
- Collections: list, tuple, set, dict
- Special: None, bytes

4. What is the difference between a list and a tuple?

List: mutable, uses []

Tuple: immutable, uses ()

5. What is a dictionary in Python?

A key-value pair collection.

Example: student = {'name': 'John', 'age': 21}

6. What are the different types of operators in Python?

- Arithmetic: +, -, *, /
- Comparison: ==, !=, >, <
- Logical: and, or, not
- Assignment: =, +=, -=
- Bitwise: &, |, ^

7. What is the difference between break, continue, and pass?

break: exits loop

continue: skips current iteration

pass: does nothing

8. Explain the use of if __name__ == '__main__':

Ensures code runs only when file is executed directly, not when imported.

9. What are *args and **kwargs?

Used to handle variable-length arguments in functions.

10. What is recursion in Python?

A function calling itself to solve smaller problems.

Example:

```
def fact(n):  
    if n==0: return 1  
  
    return n*fact(n-1)
```

11. How do you handle exceptions in Python?

Using try, except, else, finally blocks.

12. How to read and write files in Python?

Read: with open('file.txt', 'r') as f:

Write: with open('file.txt', 'w') as f:

13. What is a class and an object?

Class: blueprint for object

Object: instance of a class

14. Difference between `__init__()` and regular method?

`__init__` runs on object creation, regular methods are called manually

15. What is encapsulation?

Binding data and code together and hiding internal details

16. What is inheritance?

One class derives from another and inherits properties and methods

17. What is polymorphism?

Different classes have methods with same name but different behavior

18. What is abstraction in Python?

Hiding implementation and showing only necessary parts using abstract classes

19. What are class and instance variables?

Class: shared by all instances

Instance: unique to each object

20. What are magic methods in Python?

Special methods like `__init__`, `__str__`, etc. used internally by Python