# Python Interview Questions and Answers

**Q1. What is Python?**
Answer: Python is a high-level, interpreted, object-oriented programming language.

**Q2. What are the key features of Python?**
Answer: Easy to learn, interpreted, dynamically typed, object-oriented, extensive libraries.

**Q3. What are Python's data types?**
Answer: int, float, str, bool, list, tuple, set, dict, NoneType.

**Q4. What is the difference between list and tuple?**
Answer: Lists are mutable, tuples are immutable.

**Q5. What is the difference between set and frozenset?**
Answer: Set is mutable, frozenset is immutable.

**Q6. What is a dictionary in Python?**
Answer: A dictionary is a collection of key-value pairs, unordered and mutable.

**Q7. What is the difference between deep copy and shallow copy?**
Answer: Shallow copy copies only references, deep copy creates a new independent copy of objects.

**Q8. What is Python used for?**
Answer: Web development, data science, AI/ML, scripting, automation, desktop applications.

**Q9. What is PEP 8?**
Answer: PEP 8 is Python's style guide for writing clean and readable code.

**Q10. What are Python namespaces?**
Answer: A namespace is a container that holds variable names mapped to objects.

**Q11. What are Python scopes?**
Answer: LEGB Rule: Local, Enclosing, Global, Built-in.

**Q12. What is the difference between global and nonlocal keywords?**
Answer: global declares a variable in global scope, nonlocal declares in enclosing scope.

**Q13. What is the difference between Python 2 and Python 3?**
Answer: Python 3 uses print() function, Unicode by default, integer division behavior changed, etc.

### Q14. What are Python functions?
Answer: Functions are reusable blocks of code defined with def keyword.

### Q15. What are *args and **kwargs?
Answer: *args allows variable number of arguments, **kwargs allows variable keyword arguments.

### Q16. What are Python decorators?
Answer: Decorators are functions that modify the behavior of other functions or classes.

### Q17. What are Python generators?
Answer: Generators are functions that yield values one at a time using yield keyword.

### Q18. What are Python iterators?
Answer: Iterators implement __iter__() and __next__() methods for iteration.

### Q19. What is the difference between iterator and iterable?
Answer: Iterable is an object that can return an iterator, iterator performs actual iteration.

### Q20. What are Python comprehensions?
Answer: Compact ways to create lists, sets, and dictionaries using expressions.

### Q21. What are Python modules?
Answer: A module is a file containing Python code, functions, and classes.

### Q22. What are Python packages?
Answer: A package is a collection of modules in a directory with __init__.py.

### Q23. What is the difference between import and from import?
Answer: import imports entire module, from import imports specific attributes or functions.

### Q24. What are Python libraries?
Answer: Predefined collections of modules for specific tasks, e.g., NumPy, Pandas.

### Q25. What is the difference between is and ==?
Answer: is checks object identity, == checks value equality.

### Q26. What is mutable vs immutable in Python?
Answer: Mutable objects can change (list, dict, set), immutable cannot change (str, tuple, frozenset).

### Q27. What is Python's memory management?
Answer: Python uses automatic garbage collection with reference counting and generational GC.

### Q28. What are Python exceptions?
Answer: Errors detected during execution, handled using try-except-finally blocks.

### Q29. What is the difference between errors and exceptions?
Answer: Errors are unrecoverable issues, exceptions can be handled programmatically.

### Q30. What is Python's with statement?
Answer: with is used to manage resources like file handling using context managers.

### Q31. What is a lambda function?
Answer: Lambda is an anonymous single-line function defined using lambda keyword.

### Q32. What are Python's built-in data structures?
Answer: List, tuple, set, dictionary.

### Q33. What is the difference between append() and extend()?
Answer: append() adds single element, extend() adds elements from another iterable.

### Q34. What is the difference between remove(), pop(), and del?
Answer: remove() deletes by value, pop() deletes by index and returns element, del deletes by index.

### Q35. What is Python's pass statement?
Answer: pass is a placeholder statement that does nothing.

### Q36. What is the difference between break, continue, and pass?
Answer: break exits loop, continue skips to next iteration, pass does nothing.

### Q37. What are Python's built-in functions?
Answer: Examples: len(), type(), print(), id(), sorted(), range().

### Q38. What is Python's map function?
Answer: map() applies a function to each item of an iterable and returns an iterator.

### Q39. What is Python's filter function?
Answer: filter() filters elements based on a function that returns True or False.

### Q40. What is Python's reduce function?
Answer: reduce() applies a function cumulatively to elements (requires functools).

### Q41. What are Python's property decorators?
Answer: @property defines getter, setter, and deleter methods for class attributes.

### Q42. What are Python's class methods and static methods?
Answer: Class methods use @classmethod and take cls, static methods use @staticmethod and take no implicit arguments.

### Q43. What is multiple inheritance in Python?
Answer: A class can inherit from multiple classes.

### Q44. What is method resolution order (MRO)?

Answer: MRO is the order in which base classes are searched, follows C3 linearization.

### Q45. What is Python's super() function?

Answer: super() is used to call methods of parent class.

### Q46. What are Python's magic methods?

Answer: Special methods that start and end with __, e.g., __init__, __str__, __len__, __add__.

### Q47. What is the difference between __str__ and __repr__?

Answer: __str__ is user-friendly representation, __repr__ is developer-focused representation.

### Q48. What is pickling in Python?

Answer: Pickling is serializing objects into byte streams using pickle module.

### Q49. What is unpickling in Python?

Answer: Unpickling is deserializing byte streams back into objects.

### Q50. What are Python's virtual environments?

Answer: Virtual environments allow isolated Python environments for projects using venv or virtualenv.

### Q51. What is the GIL in Python?

Answer: Global Interpreter Lock allows only one thread to execute Python bytecode at a time.

### Q52. What are Python frameworks?

Answer: Django, Flask for web; TensorFlow, PyTorch for AI/ML.