

Lists in Python

1. **What is a list in Python?**
Answer: A list is an ordered, mutable collection of items in Python, defined using square brackets [].
 2. **How do you create a list in Python?**
Answer: `my_list = [1, 2, 3, 4, 5]`
 3. **How do you access elements in a list?**
Answer: Using indexing, e.g., `my_list[0]` for the first element.
 4. **How do you add an element to a list?**
Answer: `append()`, e.g., `my_list.append(6)`.
 5. **How do you insert an element at a specific index?**
Answer: `insert()`, e.g., `my_list.insert(2, 10)`.
 6. **How do you remove an element from a list?**
Answer: `remove()`, e.g., `my_list.remove(3)`.
 7. **How do you remove an element using an index?**
Answer: `pop()`, e.g., `my_list.pop(1)`.
 8. **How do you reverse a list?**
Answer: `my_list.reverse()`.
 9. **How do you sort a list?**
Answer: `my_list.sort()`.
 10. **How do you find the length of a list?**
Answer: `len(my_list)`.
-

Strings in Python

11. **How do you define a string in Python?**
Answer: Using single ('Hello') or double ("Hello") quotes.
12. **How do you concatenate two strings?**
Answer: Using `+`, e.g., `"Hello" + "World"`.
13. **How do you repeat a string?**
Answer: Using `*`, e.g., `"Hello" * 3`.
14. **How do you find the length of a string?**
Answer: `len(string)`.
15. **How do you convert a string to lowercase?**
Answer: `string.lower()`.
16. **How do you convert a string to uppercase?**
Answer: `string.upper()`.

17. **How do you replace a substring in a string?**

Answer: `string.replace("old", "new")`.

18. **How do you split a string?**

Answer: `string.split()`.

19. **How do you join a list of strings?**

Answer: `' '.join(list)`.

20. **How do you check if a string contains a substring?**

Answer: `"sub" in string`.

Functions in Python

21. **How do you define a function in Python?**

Answer: Using `def`, e.g.,

python

CopyEdit

```
def my_function():
```

```
    print("Hello")
```

22. **How do you call a function?**

Answer: `my_function()`.

23. **How do you pass arguments to a function?**

Answer: `def add(a, b): return a + b`.

24. **What is a default argument in Python?**

Answer: A parameter with a default value, e.g., `def greet(name="User")`.

25. **What is a lambda function?**

Answer: An anonymous function, e.g., `lambda x: x * 2`.

26. **How do you return multiple values from a function?**

Answer: Using tuples, e.g., `return a, b`.

27. **What is recursion in Python?**

Answer: A function calling itself, e.g.,

python

CopyEdit

```
def factorial(n):
```

```
    return 1 if n == 0 else n * factorial(n - 1)
```

28. **What is the difference between `*args` and `**kwargs`?**

Answer: `*args` passes multiple arguments, `**kwargs` passes key-value pairs.

29. **How do you document a function?**

Answer: Using a docstring `"""Function description"""`.

30. **What is the map() function?**

Answer: It applies a function to all elements in an iterable, e.g., `map(lambda x: x * 2, [1, 2, 3])`.

Object-Oriented Programming (OOPs)

31. **What is a class in Python?**

Answer: A blueprint for objects using `class ClassName:.`

32. **How do you create an object of a class?**

Answer: `obj = ClassName()`.

33. **What is __init__ in Python?**

Answer: A constructor method executed when an object is created.

34. **What is inheritance?**

Answer: A child class inherits from a parent class, `class Child(Parent)`.

35. **What is method overriding?**

Answer: Redefining a method in a subclass.

36. **What is encapsulation?**

Answer: Restricting access using private (`__var`) or protected (`_var`) variables.

37. **What is polymorphism?**

Answer: Using the same method name with different implementations.

38. **What is self in Python?**

Answer: Represents the instance of a class.

39. **How do you delete an object in Python?**

Answer: Using `del obj`.

40. **What is an abstract class?**

Answer: A class with at least one abstract method, defined using ABC module.

NumPy Functions

41. **How do you create a NumPy array?**

Answer: `np.array([1, 2, 3])`.

42. **How do you create a zeros array?**

Answer: `np.zeros((3,3))`.

43. **How do you create a ones array?**

Answer: `np.ones((2,2))`.

44. **How do you generate random numbers in NumPy?**

Answer: `np.random.rand(3,3)`.

45. **How do you reshape a NumPy array?**

Answer: `array.reshape(2,3)`.

Pandas Functions

46. **How do you create a Pandas DataFrame?**
Answer: `pd.DataFrame({'col1': [1,2]}).`
47. **How do you read a CSV file?**
Answer: `pd.read_csv('file.csv').`
48. **How do you get the first 5 rows of a DataFrame?**
Answer: `df.head().`
49. **How do you get column names?**
Answer: `df.columns.`
50. **How do you drop a column?**
Answer: `df.drop('column_name', axis=1).`

Matplotlib & Seaborn

51. **How do you plot a line graph using Matplotlib?**
Answer: `plt.plot(x, y).`
52. **How do you show a plot?**
Answer: `plt.show().`
53. **How do you add a title to a plot?**
Answer: `plt.title('Title').`
54. **How do you add labels to the axes?**
Answer: `plt.xlabel('X'), plt.ylabel('Y').`
55. **How do you create a Seaborn heatmap?**
Answer: `sns.heatmap(data).`
56. **How do you slice a list in Python?**
Answer: `my_list[start:end]`
57. **How do you find the index of an element in a list?**
Answer: `my_list.index(element)`
58. **How do you copy a list?**
Answer: `copy_list = my_list.copy()` or `copy_list = my_list[:]`
59. **How do you count occurrences of an element in a list?**
Answer: `my_list.count(element)`
60. **How do you merge two lists?**
Answer: `list1 + list2` or `list1.extend(list2)`

Strings in Python

61. **How do you check if a string starts with a specific substring?**

Answer: `string.startswith('sub')`

62. **How do you check if a string ends with a specific substring?**

Answer: `string.endswith('sub')`

63. **How do you remove whitespace from a string?**

Answer: `string.strip()`

64. **How do you find the position of a substring in a string?**

Answer: `string.find('sub')`

65. **How do you check if a string is numeric?**

Answer: `string.isnumeric()`

Functions in Python

66. **What is a generator function?**

Answer: A function that yields values using `yield` instead of `return`.

67. **What is the `zip()` function?**

Answer: Combines multiple iterables element-wise, e.g., `zip(list1, list2)`.

68. **What is the `filter()` function?**

Answer: Filters elements based on a condition, e.g., `filter(lambda x: x > 5, my_list)`.

69. **What is the `reduce()` function?**

Answer: Applies a function cumulatively to elements, e.g., `reduce(lambda x, y: x + y, my_list)`.

70. **What is function overloading in Python?**

Answer: Python does not support function overloading, but we can handle different argument types inside a single function.

Operators in Python

71. **What is the difference between `==` and `is` in Python?**

Answer: `==` checks value equality, `is` checks object identity.

72. **What is the `in` operator used for?**

Answer: It checks membership in a sequence, e.g., `"a" in "apple"`.

73. **What is the result of `10 / 3` and `10 // 3`?**

Answer: `10 / 3 = 3.33`, `10 // 3 = 3` (floor division).

74. **What does `**` do in Python?**

Answer: It is the exponentiation operator, e.g., `2**3 = 8`.

75. **What is the result of `bool([])`?**

Answer: `False`, because an empty list is considered `False`.

OOPs in Python

76. **What is multiple inheritance?**

Answer: A class inheriting from multiple parent classes, e.g., class C(A, B).

77. **What is method overloading?**

Answer: Defining multiple methods with the same name but different parameters (not natively supported in Python).

78. **What is a static method?**

Answer: A method that does not use self, defined using @staticmethod.

79. **What is a class method?**

Answer: A method that operates on the class rather than instances, defined using @classmethod.

80. **What is a property in Python?**

Answer: A way to define getters/setters using @property decorator.

NumPy Functions

81. **How do you create an identity matrix in NumPy?**

Answer: np.eye(n).

82. **How do you get the shape of a NumPy array?**

Answer: array.shape.

83. **How do you get the data type of a NumPy array?**

Answer: array.dtype.

84. **How do you flatten a NumPy array?**

Answer: array.flatten().

85. **How do you find the sum of all elements in a NumPy array?**

Answer: np.sum(array).

Pandas Functions

86. **How do you get summary statistics of a DataFrame?**

Answer: df.describe().

87. **How do you check for missing values in Pandas?**

Answer: df.isnull().sum().

88. **How do you fill missing values in Pandas?**

Answer: df.fillna(value).

89. **How do you drop rows with missing values?**

Answer: df.dropna().

90. **How do you group data in Pandas?**

Answer: df.groupby('column').

Matplotlib & Seaborn

91. How do you create a scatter plot using Matplotlib?

Answer: `plt.scatter(x, y)`.

92. How do you change the figure size in Matplotlib?

Answer: `plt.figure(figsize=(10,5))`.

93. How do you plot a histogram using Matplotlib?

Answer: `plt.hist(data)`.

94. How do you change the line style in a plot?

Answer: `plt.plot(x, y, linestyle='dashed')`.

95. How do you add a legend to a plot?

Answer: `plt.legend(['label1', 'label2'])`.

96. How do you change the color of a plot in Seaborn?

Answer: Using the palette parameter, e.g., `sns.barplot(x, y, palette='coolwarm')`.

97. How do you create a box plot using Seaborn?

Answer: `sns.boxplot(x, y, data=df)`.

98. How do you create a pair plot using Seaborn?

Answer: `sns.pairplot(df)`.

99. How do you create a count plot using Seaborn?

Answer: `sns.countplot(x='column', data=df)`.

100. How do you show a correlation heatmap in Seaborn?

Answer:

`python sns.heatmap(df.corr(), annot=True) plt.show()`