'''**Q1, What is NumPy, and why is it widely used in Python**

ANS

NumPy (short for Numerical Python) is a fundamental library for numerical computing in Python. It's widely used because it provides.

Why It's So Popular

A. Performance: Uses optimized C code under the hood — far faster than pure Python.

B. Foundation for other libraries: Libraries like Pandas, Scikit-learn, TensorFlow, and Matplotlib all rely on NumPy arrays.

C. Ease of use: Makes complex numerical operations more readable and concise.

D. Ecosystem: It’s the cornerstone of Python’s data science and machine learning ecosystem.

Q2.How Broadcasting Works in NumPY.

ANS.

Broadcasting applies specific rules to determine whether two arrays can be aligned for operations:

a. Check Dimensions: Ensure the arrays have the same number of dimensions or expandable dimensions.

b. Dimension Padding: If arrays have different numbers of dimensions, the smaller array is left-padded with ones.

c.Shape Compatibility: Two dimensions are compatible if:

.They are equal, or

.One of them is '''

1.
```
import numpy as np
arr = np.array([1, 2, 3])

res = arr + 1  # Adds 1 to each element
print(res)
```
'''

Q3.what is pandas dataframe.

ANS.

```
pandas DataFrame is a way to represent and work with tabular data.
It can be seen as a table that organizes data into rows and columns,
making it a two-dimensional data structure. A DataFrame can be created
from scratch,
or you can use other data structures, like NumPy arrays.
```

Q4.Explain the use of the groupby() method in Pandas.

```
ANS.
Pandas groupby() function is a powerful tool used to split a DataFrame
into groups based on one or more columns,
allowing for efficient data analysis and aggregation.
 It follows a “split-apply-combine” strategy,
 where data is divided into groups, a function is applied to each group,
and the results are combined into a new DataFrame.
For example, if you have a dataset of sales transactions,
you can use groupby() to group the data by product category and calculate
the total sales for each category.
'''
import pandas as pd
df =
pd.read_csv("https://media.geeksforgeeks.org/wp-content/uploads/nba.csv")

team = df.groupby('Team')
print(team.first()) # Let's print the first entries in all the groups
formed.
'''
```

Q5.Why is Seaborn preferred for statistical visualizations.

```
ANS.
Data visualization can be done by seaborn and it can transform complex
datasets into clear visual representations making it easier to understand,
identify trends and relationships within the data. This article will guide
you through various plotting functions available in Seaborn.
```

Q6.What are the differences between NumPy arrays and Python lists.

```
ANS.
What is a Numpy array
NumPy is the fundamental package for scientific computing in Python.
```

Numpy arrays facilitate advanced mathematical and other types of operations on large numbers of data.
Typically, such operations are executed more efficiently and with less code than is possible using Python’s built-in sequences.
Numpy is not another programming language but a Python extension module.
It provides fast and efficient operations on arrays of homogeneous data.
EX-:

```
'''
import numpy as np

a = np.array([1, 2, 3])
print(a)
'''
```

What is Python List?

A Python list is a collection that is ordered and changeable. In Python, lists are written with square brackets.
EX-:'''

```
Var = ["Geeks", "for", "Geeks"]
print(Var)
'''
```

Q7.What is a heatmap, and when should it be used.

ANS.
Heatmaps are used to show relationships between two variables, one plotted on each axis.
By observing how cell colors change across each axis,
you can observe if there are any patterns in value for one or both variables.

Q8.What does the term “vectorized operation” mean in NumPy.

ANS.
NumPy's vectorization simplifies complex computations by allowing operations directly on arrays without explicit loops.
From mathematical operations to logical comparisons and custom functions, vectorization ensures faster execution and cleaner code

Q9. How does Matplotlib differ from Plotly.

ANS.
Matplotlib: Is often preferred for academic or highly customized plots because you can fine-tune just about any aspect of the figure—fonts,

margins, axis scales, etc. Plotly: While still highly customizable, Plotly's real strength lies in interactivity and web-based visuals.22 Jan 2025

Q10.What is the significance of hierarchical indexing in Pandas.

ANS.

Hierarchical Indexing, also known as MultiIndexing, is a powerful feature in Pandas that allows you to have multiple levels of indexing on an axis (row or column). This capability is particularly useful when dealing with high-dimensional.

Q11.What is the role of Seaborn’s pairplot() function.

ANS.

Seaborn Pairplot is a Python data visualization library that allows you to plot pairwise relationships between variables within a data set.
Here's how to use it. The Seaborn Pairplot allows us to plot pairwise relationships between variables within a data set.

Q12.What is the purpose of the describe() function in Pandas

ANS.

describe()” Method. The . describe() method in Pandas is a convenient way to get a quick overview of your data. By default, it provides the count, mean, standard deviation, minimum, 25th percentile (Q1), median (50th percentile or Q2), 75th percentile (Q3),and maximum of the columns.

Q13.Why is handling missing data important in Pandas.

ANS.

n Pandas, missing values often arise from uncollected data or incomplete entries.
This article explores how to detect, handle and fill missing values in a DataFrame, ensuring clean and accurate data for analysis.
In Pandas, missing values are represented as:

Q14.What are the benefits of using Plotly for data visualization.

ANS.
It allows you to display data in a way that's easy to explore and understand, such as by zooming in,
hovering over data points for more details, and clicking to get deeper insights Plotly uses JavaScript to handle interactivity,
but you don't need to worry about that when using it in Python.

Q15. How does NumPy handle multidimensional arrays.

ANS.
In this article, we will cover the Indexing of Multi-dimensional arrays in Python using NumPy.
NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object and tools for working with these arrays.
It is the fundamental package for scientific computing with Python. It contains various features.

```
'''
# numpy library imported
import numpy as np

# creating single-dimensional array
arr_s = np.arange(5)
print(arr_s)
'''
```

Q16. What is the role of Bokeh in data visualization.

ANS.
Bokeh is an open-source python library for creating interactive visualizations that help us build beautiful charts and plots ranging from simple to complex ones.
Unlike other data visualization libraries in python like Matplotlib and seaborn,
Bokeh renders its plots using HTML and javascript.

Q17. Explain the difference between apply() and map() in Pandas

ANS.
What is Pandas apply() method

```
The apply() method can be applied both to series and Dataframes where a
function can be applied to both series and individual elements based on
the type of function provided.
'''
Syntax: s.apply(func, convert_dtype=True, args=())
'''
Pandas DataFrame apply() Method
This method can be used on both a pandas Dataframe and series. The
function passed as an argument typically works on rows/columns.
The code below illustrates how apply() method works on Pandas Dataframe.
'''
# Importing pandas library with an alias pd
import pandas as pd

# Dataframe generation
gfg_string = 'geeksforgeeks'
gfg_list = 5 * [pd.Series(list(gfg_string))]

gfg_df = pd.DataFrame(data = gfg_list)
print("Original dataframe:\n" + \
  gfg_df.to_string(index = False,
  header = False), end = '\n\n')

# Using apply method for sorting
# rows of characters present in
# the original dataframe
new_gfg_df = gfg_df.apply(lambda x:x.sort_values(), axis = 1)

print("Transformed dataframe:\n" + \
  new_gfg_df.to_string(index = False,
      header = False), end = '\n\n')
'''
```

Q18.what same advance feature of numpy.

ANS.

Key Features of NumPy

NumPy has various features that make it popular over lists.

N-Dimensional Arrays: NumPy’s core feature is ndarray, a powerful N-dimensional array object that supports homogeneous data types.

Arrays with High Performance: Arrays are stored in contiguous memory locations,

enabling faster computations than Python lists (Please see Numpy Array vs Python List for details).

Broadcasting: This allows element-wise computations between arrays of different shapes.
It simplifies operations on arrays of various shapes by automatically aligning their dimensions without creating new data.

Vectorization: Eliminates the need for explicit Python loops by applying operations directly on entire arrays.
Linear algebra: NumPy contains routines for linear algebra operations, such as matrix multiplication, decompositions, and determinants.

Q19.How does Pandas simplify time series analysis

ANS.
Although the time series is also available in the Scikit-learn library, data science professionals use the Pandas library as it has compiled more features to work on the DateTime series.
We can include the date and time for every record and can fetch the records of DataFrame.
We can find out the data within a certain range of dates and times by using the DateTime module of Pandas library.
Let's discuss some major objectives of time series analysis using Pandas library.

Q20.What is the role of a pivot table in Pandas.

ANS.
It allows you to restructure a DataFrame by turning rows into columns and columns into rows based on a specified index column, a specified columns column, and a specified values column.
This creates a summary table of the data that is easy to read and analyze.

Q21.Why is NumPy's array slicing faster than Python's list slicing.

ANS.
Performance: NumPy arrays are optimized for numerical computations, with efficient element-wise operations and mathematical functions.
These operations are implemented in C, resulting in faster performance than equivalent operations on lists.

Q22.What are some common use cases for Seaborn.

```
ANS.
Below are some common plots you can create using Seaborn:
A.Line plot. Lineplot is the most popular plot to draw a relationship
between x and y with the possibility of several semantic groupings. ...
B.Scatter Plot. ...
C.Box plot. ...
D.Violin Plot. ...
E.Swarm plot. ...
F.Bar plot. ...
G.Point plot. ...
H.Count plot.
'''
```