1.What are the new features added in Python 3.8 version?

- Assignment expressions.
- Positional-only parameters.
- Parallel filesystem cache for compiled bytecode files.
- Debug build uses the same ABI as release build.
- f-strings support = for self-documenting expressions and debugging.
- PEP 578: Python Runtime Audit Hooks.
- PEP 587: Python Initialization Configuration.

2.What are the benefits of using Python language as a tool in the present scenario?

1. Presence of third-party modules
2. Extensive support libraries(NumPy for numerical calculations, Pandas for data analytics, etc.)
3. Open source and large active community base
4. Versatile, Easy to read, learn and write
5. User-friendly data structures
6. High-level language
7. Dynamically typed language(No need to mention data type based on the value assigned, it takes data type)
8. Object-Oriented and Procedural Programming language
9. Portable and Interactive
10. Ideal for prototypes – provide more functionality with less coding
11. Highly Efficient(Python's clean object-oriented design provides enhanced process control, and the language is equipped with excellent text processing and integration capabilities, as well as its own unit testing framework, which makes it more efficient.)
12. Internet of Things(IoT) Opportunities
13. Interpreted Language
14. Portable across Operating systems

3. Give a sample code for dictionary creation.

```
# Creating a Dictionary
# with Integer Keys
Dict = {1: 'Geeks', 2: 'For', 3: 'Geeks'}
print("\nDictionary with the use of Integer Keys: ")
print(Dict)
# Creating a Dictionary
# with Mixed keys
Dict = {'Name': 'Geeks', 1: [1, 2, 3, 4]}
print("\nDictionary with the use of Mixed Keys: ")
print(Dict)
```

Output:

```
Dictionary with the use of Integer Keys:
{1: 'Geeks', 2: 'For', 3: 'Geeks'}
Dictionary with the use of Mixed Keys:
{'Name': 'Geeks', 1: [1, 2, 3, 4]}
```

4.List the Rules for creating variables in Python

Variable Names

A variable can have a short name (like x and y) or a more descriptive name (age, carname, total_volume).

Rules for Python variables:

- A variable name must start with a letter or the underscore character
- A variable name cannot start with a number
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)
- Variable names are case-sensitive (age, Age and AGE are three different variables)

Example

Get your own Python Server

```
#Legal variable names:
myvar = "John"
my_var = "John"
_my_var = "John"
myVar = "John"
MYVAR = "John"
myvar2 = "John"

#Illegal variable names:
2myvar = "John"
my-var = "John"
my var = "John"
```

5. Is NumPy a library or API?

NumPy is a Python library used for working with arrays.It also has functions for working in domain of linear algebra, fourier transform, and matrices.NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely.

NumPy stands for Numerical Python.

6. How do you manipulate values in a NumPy array

Adding / Removing Elements

Sr.No.	Element & Description
1	resize Returns a new array with the specified shape
2	append Appends the values to the end of an array
3	insert Inserts the values along the given axis before the given indices
4	delete Returns a new array with sub-arrays along an axis deleted
5	unique Finds the unique elements of an array

7. What is numerical computation with NumPy?

NumPy (numerical python) is a module which was created allow efficient numerical calculations on multi-dimensional arrays of numbers from within Python. It is derived from the

merger of two earlier modules named Numeric and Numarray. The actual work is done by calls to routines written in the Fortran and C languages.

8. Write a coding for sorting in Python?

```
# List of Integers
numbers = [1, 3, 4, 2]

# Sorting list of Integers
numbers.sort()

print(numbers)

# List of Floating point numbers
decimalnumber = [2.01, 2.00, 3.67, 3.28, 1.68]

# Sorting list of Floating point numbers
decimalnumber.sort()

print(decimalnumber)

# List of strings
words = ["Geeks", "For", "Geeks"]

# Sorting list of strings
words.sort()

print(words)
```

9.What is the difference between NumPy and pandas?

PANDAS	NUMPY
When we have to work on **Tabular data,** we prefer the p*andas* module.	When we have to work on **Numerical data**, we prefer the n*umpy* module.
The powerful tools of pandas are **Data frame and Series.**	Whereas the powerful tool of *numpy* is **Arrays.**
Pandas consume **more memory**.	*Numpy* is **memory efficient.**
Pandas has a better performance when a number of rows is **500K or more.**	*Numpy* has a better performance when number of rows is **50K or less.**
Indexing of the *pandas* series is **very slow** as compared to *numpy* arrays.	Indexing of *numpy* Arrays is **very fast**.
Pandas offer a have2d table object called **DataFrame.**	*Numpy* is capable of providing **multi-dimensional arrays.**
It was developed by Wes McKinney and was released in 2008.	It was developed by Travis Oliphant and was released in 2005.

10. Interpret on any four Methods for Series in Data Frame

Series Functions

There are some functions used in Series which are as follows:

Functions	Description
Pandas Series.map()	Map the values from two series that have a common column.
Pandas Series.std()	Calculate the standard deviation of the given set of numbers, DataFrame, column, and rows.
Pandas Series.to_frame()	Convert the series object to the dataframe.
Pandas Series.value_counts()	Returns a Series that contain counts of unique values.