**Q1. Types of Applications in Python:**

**Answer:**

Python is a multi-purpose programming language used to build various types of applications. These include web applications (using Django, Flask), desktop GUI applications (using Tkinter, PyQt), data science and machine learning tools (using Pandas, NumPy, TensorFlow), automation scripts, game development (using Pygame), and even some mobile applications (using Kivy). Python’s simplicity and vast library support make it suitable for many fields.

**Q2) What is Programming:**

**Answer**

Programming is the process of writing instructions (called code) that a computer can understand and follow to perform specific tasks. It involves using programming languages like Python, C, or Java to build software, websites, games, or automate tasks.

**Q3) What is Python:**

**Answer:**

Python is a high-level, interpreted, and general-purpose programming language developed by Guido van Rossum in 1991. It is known for its simple syntax, readability, and versatility, and is widely used in web development, data science, machine learning, automation, and more.

**Q7. How is memory managed in Python?**

**Answer:**

Memory in Python is managed automatically using the following techniques:

1. **Private Heap Space**:  
   All Python objects and data are stored in a private memory area called the *heap*. This space is managed internally by the Python memory manager.
2. **Garbage Collection**:  
   Python has an in-built garbage collector that automatically deletes unused objects and frees memory.
3. **Reference Counting**:  
   Each object keeps track of how many references are pointing to it. When this count becomes zero, the object is removed from memory.
4. **Dynamic Typing**:  
   Memory is allocated dynamically based on the type of variable during runtime.
5. **Memory Pools (Pymalloc)**:  
   For efficient memory handling, Python uses *pymalloc* to manage memory for small objects internally.

**Q8. What is the purpose of continue statement in Python?**

**Answer:**

The continue statement is used inside loops in Python. It is used to **skip the current iteration** of the loop and move to the next one without executing the remaining statements inside the loop for that particular iteration.

**Q17. What are negative indexes and why are they used?**

**Answer:**

In Python, **negative indexing** means counting elements **from the end** of a list, string, or any sequence type.

* **-1** refers to the **last element**
* **-2** refers to the **second last element**
* and so on...

**Q25) What is List? How will you reverse a list?**

**Answer:**

A **list** is a built-in data structure in Python that is used to **store multiple items in a single variable**.

* It can store **different data types**: numbers, strings, even other lists
* It is **ordered** (items have a fixed position)
* It is **mutable** (changeable

**4 Method to Reverse a List:**

1. **reverse()** – Reverses the list in-place.
2. **[::-1]** – Returns a reversed copy using slicing.
3. **reversed()** – Returns a reversed iterator, use list() to convert.
4. **Loop method** – Add each element to front of a new list.

**Q26) How will you remove last object from a list?**

**Answer:**

We use pop() to remove the last item from a list.

**Q28) Differentiate between append () and extend () methods?**

**Answer:**

**🔹 append():**

* Adds **a single item** to the end of the list.
* Even if you give a list, it will add the **whole list as one item**.

**🔹 extend():**

* Adds **multiple items** to the list.
* It **adds each element** of the other list **one by one**.

**Q30) How will you compare two lists?**

**Answer:**

We can compare two lists using the == operator.

It returns:

* True → if both lists have same elements in same order
* False → otherwise

**Q 43) What is tuple? Difference between list and tuple?**

**Answer:**

A tuple is a collection in Python that is ordered and immutable. This means the elements in a tuple cannot be changed once the tuple is created. Tuples are defined using round brackets ().

A list, on the other hand, is also an ordered collection, but it is mutable, meaning we can change, add, or remove elements after the list is created. Lists are defined using square brackets [].

**Q47) How will you create a dictionary using tuples in python?**

**Answer:**

In Python, a dictionary can be created using a list of tuples.  
Each tuple should contain exactly two elements: the first is the key, and the second is the value.

We use the dict() function to convert the list of tuples into a dictionary.

**Q51) How Do You Traverse Through a Dictionary Object in Python?**

**Answer:**

In Python, we can traverse a dictionary in the following ways:

1. By iterating through its keys.
2. By iterating through key-value pairs using the items() method.
3. By iterating through only the values using the values() method.

These methods help in accessing and processing the elements stored in a dictionary.

**Q52) How Do You Check the Presence of a Key in A Dictionary?**

**Answer:**

In Python, the presence of a key in a dictionary can be checked using the in keyword. This allows us to verify if a specific key exists in the dictionary or not. It returns True if the key is present, otherwise False.

This method is simple and efficient for checking keys in a dictionary.

**Q65) How Many Basic Types of Functions Are Available in Python?**

**Answer:**

There are two basic types of functions in Python:

1. **Built-in functions** – Predefined functions like print(), len(), type().
2. **User-defined functions** – Functions created by the user using def keyword.

**Q66) How can you pick a random item from a list or tuple?**

**Answer:**

You can use the choice() function from the random module.

python

CopyEdit

import random

random.choice(['a', 'b', 'c']) # For list

random.choice(('x', 'y', 'z')) # For tuple

**Q67) How can you pick a random item from a range?**

**Answer:**

Use random.choice() with a range() object.

python

CopyEdit

import random

random.choice(range(1, 10))

**Q68) How can you get a random number in Python?**

**Answer:**

You can use random.randint(start, end) or random.random().

python

CopyEdit

import random

random.randint(1, 100) # Random integer between 1 and 100

random.random() # Random float between 0.0 and 1.0

**Q69) How will you set the starting value in generating random numbers?**

**Answer:**

Use random.seed(value) to set the starting point (seed) for random number generation. It helps to reproduce the same random results.

python

CopyEdit

import random

random.seed(10)

**Q70) How will you randomize the items of a list in place?**

Answer:

Use random.shuffle() to shuffle the list items in place (original list will change).

python

CopyEdit

import random

my\_list = [1, 2, 3, 4]

random.shuffle(my\_list)

print(my\_list)

**Q71) What is File function in python? What are keywords to create and write file.**

**Answer:**

**1. What is a File Function in Python?**

In Python, file functions are used to perform operations on files such as creating, opening, reading, writing, and closing files. These functions help you to store data permanently in a file on your system. The main function used for handling files is open.

**2. Purpose of File Handling**

File handling is useful when you want to:

* Save data permanently (even after the program ends)
* Read data from existing files
* Add new content to a file
* Create reports, logs, and backups

**3. Important File Modes (Keywords)**

Python provides different modes that you can use with the file function depending on the task you want to perform. These modes are:

* **"r" (read):**  
  Used to open an existing file for reading. If the file does not exist, it will show an error.
* **"w" (write):**  
  Used to create a new file or overwrite an existing file. All previous content will be deleted.
* **"a" (append):**  
  Used to open a file for appending. New content is added at the end of the file without removing old data.
* **"x" (create):**  
  Used to create a new file. If the file already exists, it will show an error.
* **"t" (text):**  
  This is the default mode. Used for working with text files.
* **"b" (binary):**  
  Used for working with binary files like images, videos, etc.

**4. Writing to a File**

To write data to a file in Python, you use either:

* **"w" mode** – to create or overwrite a file with new data.
* **"a" mode** – to add (append) new data at the end of the existing file.

**5. Closing the File**After performing file operations like reading or writing, it is important to close the file. Closing the file saves the data properly and frees up system resources.

**Q83) Explain Exception handling? What is an Error in Python?**

**Answer:**

**Exception handling** in Python is a way to manage **runtime errors** (errors that occur while the program is running) without crashing the program. It allows us to respond to unexpected conditions using try, except, else, and finally blocks.

**Error** in Python is a problem in a program that can cause it to stop working. Errors are of two main types:

* **Syntax Errors** – Mistakes in the structure of code.
* **Exceptions** – Errors that occur during execution (e.g., dividing by zero, file not found, etc.)

**Q84) How many except statements can a try-except block have? Name some built-in exception classes:**

**Answer:**

A try-except block can have **multiple except statements** to handle different types of exceptions separately.

**Example:**

python

CopyEdit

try:

x = int("abc")

except ValueError:

print("Value Error occurred.")

except TypeError:

print("Type Error occurred.")

**Some built-in exception classes are:**

* ValueError
* TypeError
* ZeroDivisionError
* FileNotFoundError
* IndexError
* KeyError

**Q85) When will the else part of try-except-else be executed?**

**Answer:**

The else block is executed **only when no exception** is raised in the try block.  
It is used to write code that should run **only if everything in try is successful**.

**Q86) Can one block of except statements handle multiple exceptions?**

**Answer:**

Yes, one except block can handle **multiple exceptions** by putting them inside parentheses.

**Q87) When is the finally block executed?**

**Answer:**

The finally block is always executed **no matter what** — whether an exception occurs or not. It is used to write **cleanup code** like closing files or releasing resources.

**Q88) What happens when ‘1’ == 1 is executed?**

**Answer:**

The expression '1' == 1 returns **False** because:

* '1' is a **string**
* 1 is an **integer**

Different data types are not equal, so this comparison is **not valid** and returns **False**, but **does not cause an error**.

**Q89) How Do You Handle Exceptions with Try/Except/Finally in Python? Explain with Coding Snippet.**

You can handle exceptions using the following blocks:

* try: Write the risky code here (which might give an error).
* except: Write what to do if an error occurs.
* else: Optional – code that runs if no error occurs.
* finally: Code that always runs (optional but useful).

**Example:**

python

CopyEdit

try:

number = int(input("Enter a number: "))

result = 10 / number

print("Result is:", result)

except ZeroDivisionError:

print("You cannot divide by zero.")

except ValueError:

print("Please enter a valid number.")

else:

print("No error occurred.")

finally:

print("This will always run, no matter what.")