Python questions

1.What is python?

Python is a high-level, interpreted programming language and general purpose language known for its simplicity, readability, and versatility. It's widely used for various purposes.

- Easy to learn and write
- Clean and readable syntax
- Huge library support

2.Is python is a interpreter language?

Yes, python is a interpreter language. Python code is executed line by line by an interpreter.

3.What is the difference between the interpreter and compiler?

Languages can be classified as either interpreted or compiled. In compiled languages (like c or c ++) the entire source code into machine code at once by a compiler. The interpreter languages are executed line by line.

4.What is the difference between the python2 and python3?

The difference between the py2 and py3 is lies their syntax and underlying design. The differences are

- **Syntax differences**: print statement
  Py2: print “Hello World” in this print is a statement.
  Py3: print(“Hello World”) in this print is a function.
- **Libraries**:
  The libraries have been updated to work with py3,and some may not be compatible with the py2.
- **Default text strings:**
  Py2: Strings are stored as ASCII by default.
  Py3:Strings are stored as Unicode.

5. What is data? Data types in python?

Data refers to that can be processed and interpreted to derive meaning or information. In programming, data is represented and stored using specific data types, which dictate how the data can be manipulated and what operations can be performed on it.

Data types:

Data types are 2 they are

1. Primitive data type: it is also called as immutable data type they won’t be modify.

Ex: int, float, bool, string, none.

2. Non -primitive datatype: it is also called as mutable datatype they are modifiable. They are in-built data structures of python.
   Ex: list, tuple, dictionaries, set.

6. What is list? Give an example for that?

In python, a list is an ordered, mutable collection of items. Lists are defined by enclosing elements within square brackets [], separated by commas.

Ex: Numbers= [1,2,3,4,5]

7. What is dictionary? Give an example for that?

In python, a dictionary is a built- in data structure that stores data in key-value pairs. Each key must be unique and immutable (ex: strings, numbers, tuples), while values can be of any data type and can be duplicated. Dictionaries are defined using curly braces {}.

Ex: Details= {

“Name”:”sireesha”,

“age”:21,

“Location”:”Hyderabad”

}

8. What is tuple? Give an example for that?

A tuple in python is an ordered, immutable collection of items. This means that once a tuple is created, its elements cannot be modified, added, or removed. Tuples are defined by enclosing elements in parentheses(),with individual elements separated by commas.

Ex: abc=(10,20,30,40,50,”sireesha”)

9.what is difference between mutable and immutable? Give a single example for each of them to demonstrate.

Mutable data type: A mutable data type is one whose values can be changed.

Ex: list, dictionaries, and set

abc=[1,2,3]

abc(0)=”sireesha”

print(abc)

Immutable data type: an immutable data type is one in which the values can’t be changed or altered.

Ex: string, tuple

a=”sireesha”

a[2]=”n”

print(a)

10.What is difference between tuple and list?

Tuples are immutable data types and tuples have the fixed length, it uses round brackets or parenthesis.

Lists are mutable data types and have a dynamic length, it uses square brackets.

11. How can mutate the list?

Lists in python are mutable, their contents can be changed after they are created. Here are common ways to mutate a list is direct assignment to an index.

Ex: my_list = [1,2,3]

My_list[0]=10

12.What is the difference between the append and insert methods while mutating the list?

In python, both append() and insert() are list methods used to add elements to a list, but they differ in where the new element is placed

- Append(): This method adds a single element to the end of the list. It is a straight forward way to extend a list by adding items one by one to the last position.

  Ex: my_list=[1,2,3]

  My_list.append(4)

  Print(my_list)

  o/p:[1,2,3,4]

- Insert(): This method allows you to add a single element at a specific index within the list.
  Ex: my_list=[1,2,3]
  My_list.insert(1,99)
  Print(my_list)
  o/p:[1,99,2,3]

13. What is the difference between pop & pop index in python?

In python, there is no distinct pop_index() method. The pop() method itself handles both scenarios:

- List.pop() (without an index) when called without any arguments, pop() removes and returns the last element from the list.

```
Ex: my_list=[10,20,30,40]
Removed_item=my_list.pop()
Print(Removed_item)
Print(my_list)
```

- List.pop(index) (with an index): When an integer index is provided as an argument,pop() removes and returns the element at that specific index from the list.

```
Ex:my_list=[10,20,30,40]
Removed_item=my_list.pop(1)
Print(Removed_item)
Print(my_list)
```

14. How can you mutate the dict in python Give an example?

Dictionary is a mutable data type. The dict is a new key-value pair or adding a new key-value pair or changing the associated value for a key-value pair in the dictionary.

```
abc={
     “year”:2025
    “location”:”hyd”
}
abc[“year”]=2026
del abc[“year”]
```

15. Write nested dictionaries for electronics product?

A nested dictionary in python can be used to represent an electronics product, organizing its various attributes in a hierarchical structure.

```
electronics_products = {
  "Laptop": {
    "brand": "Dell",
    "model": "XPS 15",
    "specifications": {
      "processor": "Intel Core i7",
      "ram_gb": 16,
      "storage_gb": 512,
      "display_size_inch": 15.6
    },
    "price_usd": 1899.99,
        "availability": {
            "in_stock": True,
            "quantity": 50
        }
    },
    "Smartphone": {
        "brand": "Samsung",
        "model": "Galaxy S24 Ultra",
        "specifications": {
            "processor": "Snapdragon 8 Gen 3",
            "ram_gb": 12,
            "storage_gb": 256,
            "camera_mp": 200
        },
        "price_usd": 1199.00,
        "availability": {
          "in_stock": False,
            "estimated_restock_date": "2025-08-10"
        }
```

16. Write a list of dictionaries?

A list of dictionaries in python can be created several ways, depending on the data source and desired structure.

```
data = [
    {"name": "sireesha", "age": 21, "city": "hyderbad"},
    {"name": "srikanth", "age": 19, "city": "kurichedu"},
    {"name": "pushpanjali", "age": 21, "city": "guntur"}
]
```

17. what is operator?

In python, an operator is a special symbol or keyword that performs an operation on one or more values or variables, known as operands.

18. What are the different types of operators in python?

Python supports various types of operators, each serving a distinct purpose:

- **Arthamatic operators**: Used for performing mathematical calculations.
  Ex: +, -, *, /, %, **, //
- **Assignment operators:** Used to assign values to variables.
  Ex: =, +=, -=, *=, /=, %=, **=, //=
- **Comparison operators:** Used to compare values and return a Boolean result(true or false).
  Ex: ==, !=, >, <, >=, <=
- **Logical operators:** Used to combine conditional statements.
  Ex: and, or, not
- **Bitwise operators:** Used to perform operations on individual bits of integers.
  Ex: &, |, ^, ~, <<, >>
- **Membership operators:** Used to test if a sequence contain aspecific value.
  Ex: in, not in.

19. Create a list with length 10 and in that list 4,5 elements are 2 lists and 0 and last elements are dictionaries and remaining elements all are primitive datatypes.

```
my_list = [{1:"Siri",3:"Klp"},2.5,3,4,[10,20,30],[40,50,60],6,"Hello",8,{2:"Anjali"}]

print(type(my_list[4]))

print(type(my_list[5]))

print(type(my_list[9]))

print(type(my_list[0]))

print(type(my_list[2]))
```

20. Create a list of length 4 and every item in the list to be list again with 4 length and print first index item in all sub lists in main list?

```
 my_list=[[1,2,3,4,5],[6,7,8,9,10],[11,12,13,14,15],[16,17,18,19,20],[21,22,23,24,25]]

print(my_list[0][0])

print(my_list[1][0])

print(my_list[2][0])

print(my_list[3][0])

print(my_list[4][0])
```

21. You have ₹1500 in your wallet.

You spend:1. ₹300 on food
2. ₹250 on transport

3. ₹400 on shopping
You also earn ₹500 from freelance work.

Calculate:1.Total money spent
2.Final balance in wallet
3.Percentage of total income spent

```
My_Wallet=1500
Freelance_work=500
Total_savings=My_Wallet+Freelance_work
print("total amount:",Total_savings)


#expenses
food=300
transport=250
shopping=400
Total_expences=food+transport+shopping
print("total spent:",Total_expences)
print("final balance:",Total_savings - Total_expences)
percentage_spent=(Total_expences/Total_savings) * 100
print("Percentage:",percentage_spent)
```

22. You are playing a game. Your score changes as follows:

- **Start with score = 0**
  1.In Level 1, you gain 10 points
  2.You get a bonus of 5 points
  3.In Level 2, you lose 3 points
  4.You use a booster that doubles your score
  Show the score after each change.

```
score=0
print("start score:",score)
score+=10
print("after level 1:",score)
score+=5
print("after bonus:",score)
```

```
score-=3

print("after level 2:",score)

score*=2

print("after booster:",score)
```

23. To get admission in a college, you need at least 60% marks.

Ask the user to input their percentage. Then check:

1.Did the student qualify? (>= 60)

2.Is their percentage exactly equal to 60?
3.Did they get less than the cutoff?

```
marks=float(input("Enter marks:"))

if marks > 60:

  print("Qualified")

if marks == 60:

  print("Qualified by border")

if marks < 60:

  print("did not qualify")
```