1.What is Python? What are the benefits of using Python?

Python is a high-level, interpreted, general-purpose programming language.It was created by Guido van Rossum.

Benefits:

Python is a general-purpose programming language that has a simple, easy-to-learn syntax that emphasizes readability and therefore reduces the cost of program maintenance

Interpreted language

Dynamic typed language

2. What is a variable in Python and How do you declare a variable in Python?

A variable is a named location in the computer's memory that stores a value.

declare a variable by assigning a value to it using the assignment operator =.

Example: a= 10 //a is a variable

3. Explain the rules for naming variables in Python?

1. Valid Characters: Variable names can only contain letters (both uppercase and lowercase), numbers, and underscores. Special characters, spaces, or punctuation marks are not allowed.
2. Start with a Letter or Underscore: A variable name must begin with a letter (a-z, A-Z) or an underscore (\_). It cannot start with a number.
3. Case-Sensitive: Python is case-sensitive, so my\_variable and My\_Variable would be considered different variables.
4. Reserved Words: Avoid using Python reserved words as variable names. Examples include if, else, while, for, import, and others.
5. Descriptive and Meaningful: Choose variable names that are descriptive and convey the purpose of the variable for better code readability.

4. What are the built-in data types in Python?

* int: Integer
* float: Floating-point number
* str: String
* bool: Boolean
* list: List
* tuple: Tuple
* set: Set
* dict: Dictionary

5.Explain the difference between mutable and immutable data types with examples?

**Mutable Data Types:** Objects whose values can be changed after creation.

**Immutable Data Types:** Objects whose values cannot be changed after creation.

6.What is the purpose of the type() function in Python?

The type() function is used to get the type of an object.

Example: x=”hello python”

print(type(x))

output: <class ‘str’>

7.What is an operator in Python?

an operator is a special symbol or keyword that performs operations on variables and values.

Assignment operator(=)

Arithematic operators(+,-,\*,/,%,\*\*)

Comparison operators(<,>,<=,>=,==)

Logical operators(and,or,not)

8. What is the order of precedence in Python operators?

1. Exponentiation \*\*
2. Unary positive +, unary negation -
3. Multiplication \*, division /, modulo %
4. Addition +, subtraction -

Parentheses ( ) can be used to explicitly specify the order of evaluation, overriding the default precedence.

9. Discuss the role of arithmetic operators in Python?

Arithmetic operators in Python are used to perform mathematical operations on numeric values. These operators include addition, subtraction, multiplication, division, modulus, and exponentiation.

a+b, a-b, a\*b, a/b, a%b, a\*\*b

10. Explain the concept of modulus and its use in Python.

The modulus operator (%) in Python returns the remainder of the division of one number by another. It is often used to check if a number is even or odd

11. How do you use the and, or, and not logical operators in Python?

If we have two conditions to use logical operators.

and operator: returns true if both conditions are true

or operator: returns true if atleast one condition is true

not operator: returns opposite of given condition

12.What is a list in Python?

a list is a mutable, ordered collection of elements. Each element can be of any data type, and the elements are enclosed within square brackets [] and allows duplicates.

Example: mylist=[a,b,c,d,e,f]

13.How do you access elements in a list using indexing?

elements in a list are accessed using indexing. Indexing starts from 0 for the first element and goes up to len(list) - 1 for the last element.

Example: mylist=[a,b,c,d,e,f]

print(mylist[1]) Output: [b]

14.What is the purpose of negative indexing in Python lists?

access elements from the end of the list without the need to know its length. The last element is indexed as -1, the second-to-last as -2, and so on.

Example: mylist=[a,b,c,d,e,f]

print(mylist[-1]) Output: [f]

15.How do you add an element to the end of a list in Python?

add an element to the end of a list using the append() method. The append() method adds the specified element to the end of the list.

Example: mylist=[a,b,c,d,e,f]

mylist.append(‘g’)

print(mylist) Output: [a,b,c,d,e,f,g]

16.How can you insert an element at a specific index in a list?

 you can insert an element at a specific index in a list using the insert() method. The insert() method takes two arguments: the index at which to insert the element and the element itself.

Example: mylist=[a,b,c,d,e,f]

mylist.insert(0,”alphabet”)

Print(mylist) Output: [alphabet, a,b,c,d,e,f,g]

17.Discuss the purpose of the remove() and pop() methods in Python lists.

The remove() and pop() methods in Python lists are used for removing elements, but they differ in their functionality.

**remove() method:**

The remove() method is used to remove the first occurrence of a specified element from the list. If the element is not found, it raises a ValueError.

**pop() method:**

The pop() method is used to remove the element at a specified index. If no index is specified, it removes the last element. If the index is out of range, it raises an IndexError.

18.What is a tuple in Python?

a tuple is an ordered and immutable collection of elements. Tuples are defined using parentheses ( ).it allows duplicates because it is indexed.

Example: mytuple=(a,b,c,d,e,f)

Print(mytuple) Output: (a,b,c,d,e,f)

19. Explain the difference between lists and tuples in Python.

* List: Mutable, can be changed after creation.
* Tuple: Immutable, cannot be changed after creation.

20.What is a set in Python?

a set is an unordered collection of unique elements.

It removes duplicates ,we cannot access set elements

We can add or remove elements from set.

Example: myset={1,2,3,4,5,6,1,2}

Print(myset) Output: {1,3,4,5,2,6}

21.Discuss the difference between a set and other data types, such as lists or tuples

Uniqueness: don’t allow duplicates

Mutability: we can add or remove elements in set.

Ordering : set is unordered .

Indexing: set has no index cannot access elements and perform slicing.

22.Discuss the use of the discard() and remove() methods in Python sets.

The discard() and remove() methods are used to remove elements from a set in Python. The key difference between them is that if the element is not present in the set, discard() does nothing, while remove() raises a KeyError.

23.What are dictonaries in Python?  
Dictonaries are the datatypes in python in which the dataitems are arranged in the form of keys and values. It is opposite to sequential datatypes, where the data items were arranged by range.

#Example of a Dictionary

dict1={"name": "rahul","age": 23,"city": "kochi"}

print(dict1) output:{‘name’:’rahul’, ‘age’:23, ‘city’:’kochi’ }

24.How can you access values in a dictionary using keys?

you can access values in a dictionary using keys.

Here's an example:

student\_info = {"Name": "John", "Age": 25, "City": "New York"}

print(student\_info["Name"]) output: John

25. Explain the purpose of the update() method in Python dictionaries.

The update() method in Python dictionaries is used to merge the key-value pairs from one dictionary into another.

Example:

student\_info = {"Name": "John", "Age": 25, "City": "New York"}

additional\_info = {"Grade": "A", "Hobbies": ["Reading", "Coding"]}

student\_info.update(additional\_info)

print(student\_info)

output: {'Name': 'John', 'Age': 25, 'City': 'New York', 'Grade': 'A', 'Hobbies': ['Reading', 'Coding']}

26.Is python dictionaries mutable?  
Yes, dictionaries are mutable you can add, update, delete key, value pairs from it.

27.Can we use tuple as keys inside python dictionary?

Yes, tuple can we used as key inside python dictionary, only if it contain only string, number or tuple. If a tuple contains any mutable datatype inside it like list, it can not be used as keys.

28.Can we use lists as keys inside python dictionary?

No, python list can not be used as keys inside python dictionary, as they are mutable in nature.

29.What is .item() method inside a dictionary?  
.item() is used to arrange the complete dictionary in the form of list with tuples carrying key, value pair.

dict1 = {"name":"mradul","age":23,"city":"indore"}

print(dict1.items())

#Output : dict\_items([('name', 'mradul'), ('age', 23), ('city', 'indore')])

30.What does .pop() method do in dictionary?  
Pop inside dictionary can be used to remove the last item as well as any specific item. While in list the pop was just used to remove the last element.

example

dict1={"name": "mradul","age": 23,"city": "indore"}

print(dict1)

dict1.pop("city")

print(dict1)

#Output {'name': 'mradul', 'age': 23, 'city': 'indore'}

{'name': 'mradul', 'age': 23}

31.What is indentation in Python, and why is it important?

* Indentation refers to the spaces at the beginning of a code line. Where in other programming languages the indentation in code is for readability only, the indentation in Python is very important.
* Python uses indentation to indicate a block of code.
* Python will give you an error if you skip the indentation.

32.What is a dynamically typed language?

* Data Types are checked during execution.

Python is an interpreted language, executes each statement line by line and thus type- checking is done on the fly, during execution. Hence, Python is a Dynamically Typed Language.

33.What is a conditional statement in Python?

a conditional statement allows you to check whether a certain condition is true or false. The most common conditional statements are if, elif (else if), and else.

34. Explain the difference between the if, elif, and else statements.

if statement is used to execute a block of code only if a certain condition is true. The elif statement is used to check additional conditions if the previous conditions are false. The else statement is used to execute a block of code if none of the previous conditions are true.

# Example

number = -15

if number > 0:

print("The number is positive.")

elif number < 0:

print("The number is negative.")

else:

print("The number is zero.")

35. Explain the concept of nested if statements in Python.

A nested if statement is an if statement inside another if statement. Each if statement is executed based on the condition of the outer if statement being true.

Example:

score = 75

if score >= 60:

print("Passing grade.")

if score >= 90:

print("Excellent!")

elif score >= 80:

print("Very Good.")

elif score >= 70:

print("Good.")

else:

print("Fair.") output: Passing grade.

Good.

else:

print("Failing grade.")

36. How can you use the pass statement in an if block in Python?

the pass statement is a no-operation statement. It serves as a placeholder where syntactically some code is required but where no action is desired or necessary

37.What is the difference between *==* and *is operator* in *Python*?

Both the **==** and **is** operators in Python are used for comparison, but they function differently.

* The **==** operator checks for **value equality**.
* The **is** operator, on the other hand, validates **object identity**,

38. what is iterators in python?

An iterator is an object that can be iterated upon, meaning that you can traverse through all the values.

Example:List can be iterated

39. Discuss the role of the get() method in Python dictionaries.

The get() method is used to retrieve the value associated with a specified key.

Example:

student\_info = {"Name": "John", "Age": 25, "City": "New York"}

name = student\_info.get("Name")

print("Name:", name) output: Name:John

40. What is the purpose of the ‘ in ‘ keyword when used with strings and lists?

The in keyword in Python is used to check whether a specified element is present in a sequence (such as strings, lists, or tuples). It returns True if the element is found and False otherwise.

Example

list=[1,2,3,4,5]

check\_no=3

present= check\_no in list

print(present) output: true

41. what are the rules for local and global variables in Python?

Ans: Local variables: If a variable is assigned a new value anywhere within the function’s body, it’s assumed to be local.

Global variables: Those variables that are only referenced inside a function are implicitly global.

42. the use of // operator in Python?

Ans: It is a Floor Divisionoperator , which is used for dividing two operands with the result as quotient showing only digits before the decimal point.

Example: 10//5 = 2

43.How can you check if all the characters in a string are alphanumeric?

You can use the isalnum() method, which returns True if all the characters are alphanumeric, meaning alphabet letter (a-z) and numbers (0-9).

44. Explain the characteristics of the str data type in Python.

str data type is used to store textual data. Strings are immutable, meaning their values cannot be changed after creation. They support various operations, including indexing, slicing, and string concatenation.

45. Explain the concept of type casting in Python?

Type casting in Python refers to the process of converting a variable from one data type to another.

Example

mytuple=(1,2,3)

mylist=list(mytuple)

print(mylist) output: [1,2,3]

46. what is the use of copy() method in python?

The copy() method in Python sets is used to create a shallow copy of the set. A shallow copy creates a new set. but the elements themselves are still references to the same objects as the original set.

47. Explain the concept of a frozen set in Python.

A frozen set is an immutable version of set. Once created, a frozen set cannot be modified by adding or removing elements.

48.What is Inheritance in Python?

* Inheritance allows us to define a class that inherits all the methods and properties from another class.
* Parent class is the class being inherited from, also called base class.
* Child class is the class that inherits from another class, also called derived class.

49.What is a class in Python, and how do you use it?

* A Class is like an object constructor, or a "blueprint" for creating objects.
* You can create a class with the class keyword:

Example class MyClass:  
 x = 5

50.Can you copy a List in Python by simply writing: list2 = list1?

* No, because: list2 will only be a *reference* to list1, and changes made in list1 will automatically also be made in list2.
* To make a copy of a list, you can use copy() or the list() method.