

UNIT V PART_A

FILES, MODULES AND PACKAGES

1. What is module and package in Python?

In Python, module is the way to structure program. Each Python program file is a module, which imports other modules like objects and attributes.

2. Explain how can you access a module written in Python from C?

We can access a module written in Python from C by following method, Module =
=PyImport_ImportModule(—<modulename>||);

3. Mention five benefits of using Python?

- Python comprises of a huge standard library for most Internet platforms like Email, HTML, etc.
- Python does not require explicit memory management as the interpreter itself allocates the memory to new variables and free them automatically Provide easy readability due to use of square brackets
- Easy-to-learn for beginners
- Having the built-in data types saves programming time and effort from declaring variables

4. How to open a new file in Python?

Opening a file creates a file object. In this example, the variable f refers to the new file object. >>> f = open("test.dat", "w")
>>> print f<open file _test.dat', mode _w' at fe820> The open function takes two arguments. The first is the name of the file, and the second is the mode. Mode "w" means that we are opening the file for writing.

5. Explain how the write method works on a file.

```
>>> f.write("Now is the time")  
>>> f.write("to close the file")
```

Closing the file tells the system that we are done writing and makes the file available for reading:

```
>>> f.close()
```

6. Which method is used to read the contents of a file which is already created?

The read method reads data from the file. With no arguments, it reads the entire contents of the file:

```
>>> text = f.read()  
>>> print text
```

Now is the timeto close the file

7. What is a text file? Give an example for a text file.

A text file is a file that contains printable characters and whitespace, organized into lines separated by newline characters.

To demonstrate, we'll create a text file with three lines of text separated by newlines:

```
>>> f = open("test.dat", "w")
>>> f.write("line one\nline two\nline three\n")
>>> f.close()
```

8. What is the difference between break and continue statement?

The break statement is new. Executing it breaks out of the loop; the flow of execution moves to the first statement after the loop.

The continue statement ends the current iteration of the loop, but continues looping. The flow of execution moves to the top of the loop, checks the condition, and proceeds accordingly.

9. What is meant by directory? How and where is it useful?

When we want to open a file somewhere else, you have to specify the path to the file, which is the name of the directory (or folder) where the file is located:

```
>>> f = open("/usr/share/dict/words", "r")
>>
>
printf.readline()
Aarhus
```

This example opens a file named words that resides in a directory named dict, which resides in share, which resides in usr, which resides in the top-level directory of the system, called .

10. Explain pickling and how import pickle works.

Pickling is so called because it —preserves data structures. The pickle module contains the necessary commands. To use it, import pickle and then open the file in the usual way:

```
>>> import pickle
>>> f = open("test.pck", "w")
```

11. What is an exception? Explain with few examples.

Whenever a runtime error occurs, it creates an exception. Usually, the program stops and Python prints an error message.

Forexample,

dividing by zero creates an exception:

```
>>> print 55/0
```

ZeroDivisionError: integer

division or modulo So does

accessing a nonexistent list item:

```
>>> a = []
```

```
>>> print a[5]
```

12. What is module and package in Python?

In Python, module is the way to structure program. Each Python program file is a module, which imports other modules like objects and attributes.

13. Explain how can you access a module written in Python from C?

We can access a module written in Python from C by following method, Module =
=PyImport_ImportModule(—<modulename>||);

14. Mention five benefits of using Python?

- Python comprises of a huge standard library for most Internet platforms like Email, HTML, etc.
- Python does not require explicit memory management as the interpreter itself allocates the memory to new variables and free them automatically Provide easy readability due to use of square brackets
- Easy-to-learn for beginners
- Having the built-in data types saves programming time and effort from declaring variables

15. How to open a new file in Python?

Opening a file creates a file object. In this example, the variable f refers to the new file object. >>> f = open("test.dat", "w")
>>> print f<open file _test.dat', mode _w' at fe820> The open function takes two arguments. The first is the name of the file, and the second is the mode. Mode "w" means that we are opening the file for writing.

16. Explain how the write method works on a file.

```
>>> f.write("Now is the time")
```

```
>>> f.write("to close the file")
```

Closing the file tells the system that we are done writing and makes the file available for reading:

```
>>> f.close()
```

17. Which method is used to read the contents of a file which is already created?

The read method reads data from the file. With no arguments, it reads the entire contents of the file:

```
>>> text = f.read()
```

```
>>> print text
```

Now is the timeto close the file

18. What is a text file? Give an example for a text file.

A text file is a file that contains printable characters and whitespace, organized into lines separated by newline characters.

To demonstrate, we'll create a text file with three lines of text separated by newlines:

```
>>> f = open("test.dat", "w")
>>> f.write("line one\nline two\nline three\n")
>>> f.close()
```

19. What is the difference between break and continue statement?

The break statement is new. Executing it breaks out of the loop; the flow of execution moves to the first statement after the loop.

The continue statement ends the current iteration of the loop, but continues looping. The flow of execution moves to the top of the loop, checks the condition, and proceeds accordingly.

20. What is meant by directory? How and where is it useful?

When we want to open a file somewhere else, you have to specify the path to the file, which is the name of the directory (or folder) where the file is located:

```
>>> f = open("/usr/share/dict/words", "r")
>>
>
printf.readline()
Aarhus
```

This example opens a file named words that resides in a directory named dict, which resides in share, which resides in usr, which resides in the top-level directory of the system, called .

21. Explain pickling and how import pickle works.

Pickling is so called because it —preserves data structures. The pickle module contains the necessary commands. To use it, import pickle and then open the file in the usual way:

```
>>> import pickle
>>> f = open("test.pck", "w")
```

22. What is an exception? Explain with few examples.

Whenever a runtime error occurs, it creates an exception. Usually, the program stops and Python prints an error message.

Forexample,

dividing by zero creates an exception:

```
>>> print 55/0
ZeroDivisionError: integer
```

division or modulo So does

accessing a nonexistent list item:

```
>>> a = []
>>> print a[5]
```

IndexError: list index out of range

Or accessing a key that isn't in the dictionary:

```
>>> b = {}
>>> print
b['what']
KeyError:
what
```

23. List some few common Exception types and explain when they occur.

- ArithmeticError- Base class for all errors that occur for numeric calculations.
- OverflowError- Raised when a calculation exceeds maximum limit for a numeric type.
- ZeroDivisionError- Raised when division or modulo by zero takes
 - place.
- ImportError- Raised when an import statement fails.
- IndexError- Raised when an index is not found in a sequence.
- RuntimeError- Raised when a generated error does not fall into any category.

24. Write a simple program which illustrates Handling Exceptions. w

```
x=int(input(—Please
enter a number:!))
break
except ValueError:
print(—Oops! That was no valid number. Try again...!)
```

25. What are packages in Python?

A package is a collection of Python modules. Packages allow us to structure a collection of modules. Having a directory of modules allows us to have modules contained within other modules. This allows us to use qualified module names, clarifying the organization of our software.

26. Explain what is meant by namespaces and scoping.

Variables are names or identifiers that map to objects. A namespace is a dictionary of variable names/keys and their corresponding object values. A python statement can access variables in a local namespace and global namespace. If the local and global variables have the same name, the local variable shadows the global variable. Each function has its own local namespace.

PART B

- 1) Answer the following questions.
 - a) Write a small code to illustrate try and except statements in Python. (4 marks)
 - b) What are packages? Give an example of package creation in Python. (4 marks)
 - c) Compare and contrast Extending and Embedding Python. (4 marks)
 - d) Write an algorithm to check whether a student is pass or fail, the total marks of student being the input. (4 marks)

- 2) Answer the following questions.
 - a) Write a program to enter a number in Python and print its octal and hexadecimal equivalent. (6 marks)
 - b) Demonstrate the use of Exception Handling in Python.(10 marks)

- 3) Answer the following questions.
 - a) What are modules in Python? Explain. (4 marks)
 - b) Explain in details about namespaces and scoping. (8 marks)
 - c) Explain about the import statement in modules. (4 marks)

- 4) Answer the following questions.
 - a) Explain about the different types of Exceptions in Python. (6 marks)
 - b) Describe about Handling Exceptions in detail with examples. (10 marks)

- 5) Explain in detail about Python Files, its types, functions and operations that can be performed on files with examples. (16 marks)