

# Answers of Question Bank

## Class XI

### Informatics Practices

(Book Reference : Sumita Arora)

#### Chapter – 1: BASIC COMPUTER ORGANIZATION

##### Very Short answer Type Questions

**Q.1 What is volatile memory?**

**Ans:** RAM is known as Volatile Memory because when we switch off the computer its data is vanished.

**Q.2 Define each of the following:**

**(a) byte      (b) kilobyte   (c) megabyte                      (d) gigabyte                      (e) terabyte**

**Ans:** (a) byte: This is the unit of memory in computer. 1 byte = 8 bits

(b) kilobyte: This is the unit of memory in computer. 1 kilobyte = 1024 bytes

(c) megabyte: This is the unit of memory in computer. 1 megabyte = 1024 kilobytes

(d) gigabyte: This is the unit of memory in computer. 1 gigabyte = 1024 megabytes

(e) terabyte: This is the unit of memory in computer. 1 terabyte = 1024 gigabytes.

##### Short Answer Type Questions

**Q.1 State the basic units of computer. Name the subunits that make up the CPU, and give the function of each of the unit.**

**Ans:** Basic units of computer are Input Unit, Central Processing Unit and Output Unit. Sub units of CPU are Arithmetical Logical Unit(ALU), Control Unit (CU) and Memory Unit(MU).

**Q.2 What is the function of memory? What are its measuring units?**

**Ans:** The computer memory is a temporary storage area. It holds the data and instructions that the Central Processing Unit (CPU) needs. Before a program can run, the program is loaded from some storage medium into the memory. This allows the CPU direct access to the program. Its measuring units are byte, kilobyte, megabyte, gigabyte, terabyte etc.

**Q.3 Why is primary memory termed as 'destructive write' memory but 'non-destructive read' memory?**

**Ans:** The primary memory is called destructive write because the data enter here are temporary. That's why your RAM gets cleared after every restart.

**Q.4 What is the role of CPU of a mobile system?**

**Ans:** A mobile processor is found in mobile computers and cellphones.

A CPU chip is designed for portable computers, it is typically housed in a smaller chip package, but more importantly, in order to run cooler, it uses lower voltages than its desktop counterpart and has more sleep mode capability. A mobile processor can be throttled down to different power levels or sections of the chip can be turned off entirely when not in use. Further, the clock frequency may be stepped down under low processor loads. This stepping down conserves power and prolongs battery life.

**Q.5 What is SoC? how it is different from CPU? Why is it considered a better development?**

**Ans:** A system on a chip (SoC) combines the required electronic circuits of various computer components onto a single, integrated chip (IC). SoC is a complete electronic substrate system that may contain analog, digital, mixed-signal or radio frequency functions. Its components usually include a graphical processing unit (GPU), a central processing unit (CPU) that may be multi-core, and system memory (RAM).

Because SOC includes both the hardware and software, it uses less power, has better performance, requires less space and is more reliable than multi-chip systems. Most system-on-chips today come inside mobile devices like smartphones and tablets. These are considered a better development because of their small size and speed capability.

**Q.6 What are various categories of software?**

**Ans:** Software are classified into following categories –

- (i) System Software
  - a. Operating System
  - b. Language Processor
- (ii) Application Software
  - a. Packages
  - b. Utilities
  - c. Customized software
  - d. Developer Tools

**Q.7 What is the difference between an interpreter and a compiler?**

**Ans: Interpreter:** Interpreter is a type of system software that translates and executes instructions written in a computer program line-by-line, unit by unit etc. It is slower in execution because each time when you run the program translation is required.

**Compiler:** Compiler is another type of system software that translates and executes instructions written in a computer program in one go. Once compiled program need not to translate again so it works faster.

**Q.8 What is application software? Why it is required?**

**Ans:** Application software is the set of programs necessary to carry out operations for a specific task. Such as for word processing there are many application software like MS-Word, Wordpad etc. These software are required to perform special task using the computer like painting, recording, typing, data handling etc.

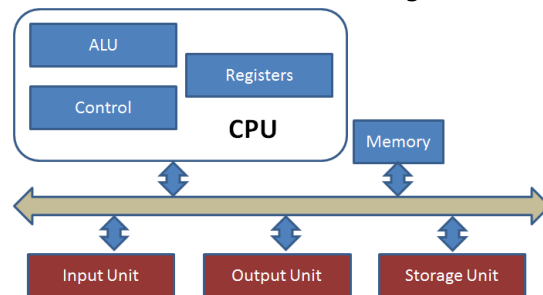
**Q.9 Briefly explain the basic architecture of a computer.**

**Ans:** Computer organization refers to logical structure of a computer describing how its components are connected to one another, how they affect one another's functioning and contributes to overall performance of computer.

Computers follow the 'IPO' principal i.e.

Input → Process → Output

(That means a certain input is processed to Generate specific output)

**Q.10 What do you understand by input unit? What is its significance? What does computer system consist of?**

**Ans:** Input unit is formed by the input devices (Keyboard, mouse, MICR, OBCR etc.) attached to the computer. Input unit is responsible for taking input and converting it into computer understandable form (the binary code). Some common input devices are:

- (i) Keyboard
- (ii) Mouse
- (iii) Microphone
- (iv) Scanner
- (v) Webcam
- (vi) Optical Bar Code Reader
- (vii) Optical Mark Reader
- (viii) Magnetic Ink Character Reader
- (ix) Touch Pad
- (x) Track Ball
- (xi) Joy stick
- (xii) Touch Screen
- (xiii) Biometric Sensors. Etc.

**Q.11 What is the function of CPU in a computer system? What are its sub units?**

**Ans:** The central processing unit (CPU) of a computer is a piece of hardware that carries out the instructions of a computer program. It performs the basic arithmetical, logical, and input/output operations of a computer system. The CPU is like the brains of the computer - every instruction,

no matter how simple, has to go through the CPU. So let's say you press the letter 'k' on your keyboard and it appears on the screen - the CPU of your computer is what makes this possible. The CPU is sometimes also referred to as the central processor unit, or processor for short. So when you are looking at the specifications of a computer at your local electronics store, it typically refers to the CPU as the processor. Its sub units are:

(i) Control Unit                      (ii) Arithmetical and Logical Unit (ALU)                      (iii) Memory Unit

**Q.12 What functions are performed by the control unit?**

**Ans:** The CU controls and guides the interpretation, flow and manipulation of all data and information. CU sends control signals until the required operations are done properly by ALU and memory. Another function of CU is the program execution. It means carrying out all the instructions stored in the program. CU gets program instruction from the memory and executes them one after the other. CU acts as a supervisor by controlling and guiding the operation taking place.

**Q.13 Distinguish between CPU and ALU?**

**Ans:** Difference Between **ALU and CPU** is that arithmetic logic unit (ALU), another component of the processor, performs arithmetic, comparison, and other operations. While Processor also central processing unit (CPU), interprets and carries out the basic instructions that operate a computer.

The main difference between CPU and ALU is that the CPU is an electronic circuit that handles instructions to operate the computer while the ALU is a subsystem of the CPU that performs arithmetic and logical operations.

**Q.14 What is the function of output unit in a computer system?**

**Ans:** Input devices are the hardware that give computers instructions. Output devices relay the response from the computer in the form of a visual response (monitor), sound (speakers) or media devices (CD or DVD drives). The purpose of these devices is to translate the machine's response to a usable form for the computer user.

**Q.15 Distinguish between internal and external memory.**

**Ans:** Internal memory is usually chips or modules that you attach directly to the motherboard. Internal Memory is a circular disc that continuously rotates as the computer accesses its data. External memory often comes in the form of USB flash drives; CD, DVD, and other optical discs; and portable hard drives.

**Q.16 What are RAM and ROM? How are they alike? How are they different? What are PROM, EPROM, EEPROM?**

**Ans:** A ROM chip is a non-volatile storage medium, which means it does not require a constant source of power to retain the information stored on it.

A RAM chip is volatile, which means it loses any information it is holding when the power is turned off.

Both of them are known as primary memory as they can directly work with CPU.

Read Only Memory (ROM)

Programmable Read Only Memory (PROM)

Erasable Programmable Read Only Memory (EPROM)

Electrically Erasable Programmable Read Only Memory (EEPROM)

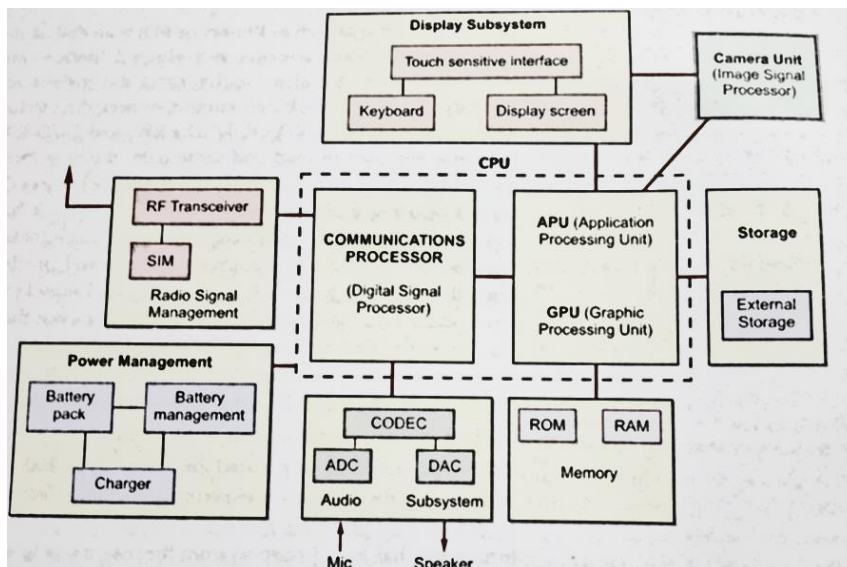
**Q.17 What are major functional components of a mobile system?**

**Ans:** Major functional components of a mobile system are-

- (i) Mobile Processor
  - a. Communications Processing Unit
  - b. Application Processing Unit
  - c. GPU (Graphics Processing Unit)
- (ii) SoC (System on a chip)
- (iii) Display Subsystem
  - a. Display Screen
  - b. Touch Sensitive Interface
  - c. Touch Sensitive Keyboards
- (iv) Camera Subsystem
- (v) Mobile System Memory
  - a. RAM
  - b. ROM
- (vi) Storage
- (vii) Power Management Subsystem

**Q.18 Draw a block diagram depicting organization of a mobile system.**

**Ans:**

**Q.19 What is the role of Power Management Unit in a mobile system?**

**Ans:** This subsystem is responsible for providing power to a mobile system. The mobile systems work on limited power provided through an attached battery unit. This system has a battery management system that works with a battery charger and a battery unit and provides power to the mobile system in required form.

It also contains a collection of different functions like battery charging, monitoring and supplying many different voltages these systems require. It also contains software controlled turn-on and turn-off feature to optimize the power consumption and battery life.

**Q.20 What does the communication processor do?**

**Ans:** this subsystem is responsible for making and receiving phone calls on a mobile handset. It has a digital signal processor that helps it work with RF Transceiver and the Audio subsystem.

**Q.21 What is GPU? How is it useful?**

**Ans:** Graphics Processing Unit assists the CPU by handling the visuals, other graphically-rich applications. In short, GPU handles all graphics-related chores of a mobile CPU.

**Q.22 What is system software?**

**Ans:** The software that controls internal computer operations is called system software. It manages all the resources of a system. Its example is Operating System.

**Q.23 Discuss the role of utility software in the context of computer performance?**

**Ans:** Utilities are those application programs that assist the computer by performing housekeeping functions like backing up disk or scanning/cleaning viruses or arranging information etc. its example is Antivirus software.

**Q.24 What is the importance of an OS?**

**Ans:** An operating system is a program which acts as an interface between a user and the hardware. It manages all the resources of the computer system. It provides an environment to the user to work with. All the applications are installed after the operating system is installed. It manages memory, processing, storage, memory etc.

**Q.25 How are software libraries useful? Name some software libraries of Python.**

**Ans:** A software library is a predefined and available to use, suit of data and programming code in the form of prewritten code/ functions/scripts/classes etc. that can be used in the development of the new software programs and applications.

Some software library in python are:

- (i) NumPy (numerical Python)
- (ii) SciPy (Scientific Python)
- (iii) Pandas Library

**Q.26 What is the utility of these software?**

**(a) disk fragmentor**

**(b) backup software**

**Ans:** **(a) disk fragmentor:** A file is fragmented when it becomes too large for your computer to store in a single location on a disk. When this happens, your computer splits the file up and stores in pieces. You can use fragmented files, but it takes your computer longer to access them.

**(b) Backup software:** This utility program facilitates the backing-up of disk. Back-up means duplicating the disk information so that in case of any damage or data-loss, this backed-up data may be used.

## Chapter – 2: GETTING STARTED WITH PYTHON

### Very Short answer Type Questions

**Q.1 When was Python released?**

**Ans-** Python was released in February 1991.

**Q.2 Who developed Python?**

**Ans-** Python was developed by Guido Van Rossum.

**Q.3 Which two languages contributed to Python as a Programming Language?**

**Ans-** ABC language and Modula 3.

**Q.4 Is Python an Object Oriented Language?**

**Ans-** Yes, Python is an object oriented language.

**Q.5 Python got its name from which show?**

**Ans-** Python got its name from famous BBC comedy show "Monty Python's Flying Circus".

**Short Answer Type Questions****Q.1 'Python is an interpreted language'. What does it mean to you?**

**Ans-** It means that the Python installation interprets and executes the code line by line at a time.

**Q.2 What does a cross platform language mean?**

**Ans-** it means a language can run equally on variety of platforms-Windows, Linux/UNIX, Macintosh, Supercomputers, Smart phones etc.

**Q.3 Python is a Free and Open Source language. What do you understand by this feature?**

**Ans-** It means Python is freely available without any cost. Its source code is also available. One can modify, improve/extend an open source software.

**Q.4 In how many ways, can you work in Python?**

**Ans-**In two ways we can work in Python-

- (i) Interactive mode
- (ii) Script Mode

**Q.5 What is the difference between Interactive mode and Script Mode in Python?**

**Ans-** In interactive mode, one command can run at a time and commands are not saved. Whereas in Script mode, we can save all the commands in the form of a program file and can see output of all lines together.

**Q.6 What are the advantages of Python?**

**Ans-** Advantages-

- i. Easy to use OO Language.
- ii. Expressive Language.
- iii. Interpreted Language.
- iv. It is complete.
- v. Cross-Platform Language.
- vi. Free and Open Source.
- vii. Variety of Usage/ Applications

**Q.7 What are the limitations of Python?**

**Ans-** limitations-

- i. Not the fastest language.
- ii. Lesser Libraries than C, Java, Perl.
- iii. Not strong on Type Binding.
- iv. Not easily convertible.

**Skill Based Questions****Q.1 Which of the following are not valid strings in Python?**

(a)"Hello" (b)'Hello' (c)"Hello" (d)'Hello" (e){Hello}

**Ans-** String (c) , (d) and (e) are not valid strings.

**Q.2 Write Instructions in python to get the following result: (Do it in both interactive mode and script mode)**

***I am a student of KV Barabanki***

***I live in Barabanki***

***And I love Barabanki.***

***Barabanki is 20 KM away from Lucknow***

***This Place is famous for Dewa Sharif***

**Ans-** In Interactive Mode-

```
>>> print("I am a student of KV Barabanki \nI live in Barabanki \nAnd I love Barabanki\n\n Barabanki is 20 KM away from Lucknow\n This Place is famous for Dewa Sharif")
```

**In Script Mode-**

```
print("I am a student of KV Barabanki \n I live in Barabanki \nAnd I love Barabanki\n\n Barabanki is 20 KM
away from Lucknow\n This Place is famous for Dewa Sharif")
```

(save this code to a file and then run the file)

## Chapter – 3: PYTHON FUNDAMENTALS

### Very Short answer Type Questions

**Q.1 What is None literal in Python?**

**Ans:** Python has one special literal, which is **None**. The None literal is used to indicate absence of value. It is also used to indicate the end of lists in Python. It means "There is nothing here".

**Q.2 What is the error in following code: `x, y = 7` ?**

**Ans:** The following error comes - 'int' object is not iterable. Which means an integer object i.e. cannot be repeated for x and y. one more integer object is required after 7.

**Q.3 what will the following code do: `a=b=18` ?**

**Ans:** This code will assign 18 to a and b both.

**Q.4 Following code is creating problem `X = 0281`, find reason.**

**Ans:** 0281 is an invalid token.

**Q.5 Find the error in the following code:**

(a) `temp=90`  
`Print temp`

(b) `a=12`  
`b = a + b`  
`print( a And b)`

(c) `print("x="x)`

(d) `a, b, c=2, 8, 4`  
`print(a, b, c)`  
`c, b, a = a, b, c`  
`print(a; b; c)`

(e) `x = 23`  
`4 = x`

(f) `else = 21-4`

**Ans:** (a) Missing parentheses in call to 'print'.  
(b) Name 'b' is not defined.  
(c) Invalid Syntax.  
(d) Invalid Syntax in second print statement.  
(e) can't assign to literal in second line.  
(f) Invalid Syntax.

**Q.6 Find the error in the following code:**

(a) `y = x +5`  
`print(x,y)`

(b) `a=input("Value: ")`  
`b = a/2`  
`print( a, b)`

(c) `print(x = y = 5)`

**Ans:** (a) Name 'x' is not defined.  
(b) Unsupported operand type(s) for /: 'str' and 'int'.  
(c) Invalid Syntax.

### Short Answer Type Questions

**Q.1 What is the difference between a keyword and an identifier?**

**Ans: Difference between Keyword and Identifier:** Every language has keywords and identifiers, which are only understood by its compiler. Keywords are predefined reserved words, which possess special meaning. An identifier is a unique name given to a particular variable, function or label of class in the program.

**Q.2 What are literals in Python? How many types of Literals allowed in Python?**

**Ans: Literals:** Python comes with some built-in objects. Some are used so often that Python has a quick way to make these objects, called literals.

The literals include the string, Unicode string, integer, float, long, list, tuple and dictionary types.

**Q.3 How many types of sequences are supported in Python?**

**Ans:** Three Types of Sequences are supported in python:

- (i) String
- (ii) List
- (iii) Tuple

**Q.4 What factors guide the choice of identifiers in program?**

- Ans:**
- (i) An identifier must start with a letter or underscore followed by any number of digits and/or letters.
  - (ii) No reserved word or standard identifier should be used.
  - (iii) No special character (Other than underscore) should be included in the identifier.

**Q.5 What is the difference between an expression and a statement in Python?**

**Ans:** A statement is an instruction that the Python interpreter can execute. We have only seen the assignment statement so far. Some other kinds of statements that we'll see shortly are while statements, for statements, if statements, and import statements. (There are other kinds too!)

An expression is a combination of values, variables, operators, and calls to functions. Expressions need to be evaluated. If you ask Python to print an expression, the interpreter evaluates the expression and displays the result.

**Q.6 What are tokens in Python? How many types of tokens allowed in Python?**

**Ans:** Tokens are the smallest unit of the program. There are following tokens in Python:

- Reserved words or Keywords
- Identifiers
- Literals
- Operators
- Punctuators

*Definition of all tokens may come. Which is not given in this question bank.*

**Q.7 What are operators? What is their function? Give examples of some unary and binary operators.**

**Ans:** "Operators are those symbols used with operands, which tells compiler which operation is to be done on operands." in other words – "operators are tokens that trigger some computation/action when applied to variables and other objects in an expression."

Operators are of following types:

- **Unary operators** like (+) Unary Plus, (-) Unary Minus, not etc.
- **Binary Operators** like (+) addition, (\*) multiplication, and etc.
- 

**Q.8 What is block/code block/suit in Python?**

**Ans:** Sometimes a group of statements is part of another statement of function. Such a group of one or more statements is called **block** or **code-block** or **suit** in python. e.g.

```
if a>b:
    print("A is graeter")
    print("Value of A is : ",a)
else:
    print("B is graeter")
    print("Value of A is : ",b)
```

Here both the sections are separate code-blocks

**Q.9 What is the role of indentation in Python?**

**Ans:** Indentation plays a very important role in Python. Python uses indentation to create blocks of code. Statements at same indentation level are part of same block/suit. You cannot unnecessarily indent a statement; python will raise an error for that.



**Q.10 How many types of strings are supported by Python?****Ans:** Python supports two types of strings:

- (i) Single-line string That terminates in single line.
- (ii) Multi-line String That stores multiple lines of text.

**Skill Based Questions****Q.1 What will be the sizes of following constants?****(a) '\a' (b) "\a" (c) "Kumar\s" (d) \"'\" (e) "it's"****Ans:** (a) 50 (b) 50 (c) 56 (d) 50 (e) 53

This screenshot is the output of the above question.

```
>>> sys.getsizeof(a)
28
>>> sys.getsizeof('\a')
50
>>> sys.getsizeof("\a")
50
>>> sys.getsizeof("kumar\s")
56
>>> sys.getsizeof(\"'\")
50
>>> sys.getsizeof("it's")
53
```

**Q.2 How can you create multi-line strings in Python?****Ans:** We can create multi-line string by putting a backslash (\) at the end of line which allows you to continue typing in next line in same string.

```
>>> Text1="Hello\
World"
>>> Text1
'HelloWorld'
```

**Q.3 Which of the following are syntactically correct strings? State reason.**

- (a) "Python is nice Language"
- (b) 'He called me "Friend!" when he came'
- (c) "Very Good'
- (d) 'This is a good book'
- (e) "Namaste
- (f) "I liked 'Harry Potter' very much"

**Ans:** (a) Correct (b) Correct (c) Incorrect (d) Correct (e) Incorrect (f) Correct**Q.4 What is the error in following Python program with one statement?**

```
print("My name is : ", name)
```

**suggest a solution****Ans:** Error is : "name 'name' is not defined". And the solution is to declare the variable-name before this statement.

```
>>> name="aa"
>>> print("My name is : ", name)
My name is : aa
```

**Q.5 Predict the output of the following:**

```
x,y=7,2
x,y,x=x+1,y+3,x+10
print(x,y)
```

**Ans:** Output: 17 5**Q.6 What will be the output of the following code:**

```
name='Hari'
age=18
print(name," , you are ",age," now but ",end="")
print("You will be ",age+1," next Year")
```

**Ans:** Output: Hari , you are 18 now but You will be 19 next year

**Q.7 Write a Program to obtain temperature in Celsius and convert it into Fahrenheit using formula –**

$$C \times 9/5 + 32 = F$$

```
c=int(input("Enter the value in Celsius"))
f=(c * 9/5) + 32
```

**Ans:** `print("Temperature in fahrenheit is : ",f)`

**Q.8 Predict output:**

```
a,b,c=2,3,4
a,b,c=a*a,a*b,a*c
print(a,b,c)
```

**Ans:** Output:        4        6        8

**Q.9 WAP to read today's date (only date Part) from user. Then display how many days are left in the current month.**

**Ans:**

```
import datetime
td=0
now=datetime.datetime.now()
print(now.day)
if now.month==2:
    td=28
elif now.month in(1,3,5,7,8,10,12):
    td=31
else:
    td=30
print("Total remaining days in the current month are : ", td-now.day)
```

**Q.10 WAP to print the area of circle when radius of the circle is given by user.**

**Ans:**

```
r=int(input("Enter the radius : "))
area= 3.14*r*r
print("Area of circle is : ",area)
```

**Q.11 WAP to print the volume of a cylinder when radius and height of the cylinder is given by user.**

**Ans:**

```
r=int(input("Enter the radius : "))
h=int(input("Enter the height : "))
vol=3.14*r*r*h
print("Volume of Cylinder is : ",vol)
```

**Q.12 WAP that asks your height in centimeters and converts it into foot and inches.**

**Ans:**

```
cm=int(input("Enter height in Centimeters : "))
foot=cm//30
rcm=cm%30
inches=rcm*0.393701
print("Height is : ",foot," Foot ",inches," Inches ")
```

**Q.13 WAP to find area of a triangle.**

**Ans:**

```
import math
a=int(input("Enter side 1 on triangle : "))
b=int(input("Enter side 2 on triangle : "))
c=int(input("Enter side 3 on triangle : "))
s=(a+b+c)/2
area=s*math.sqrt((s-a)*(s-b)*(s-c))
print("Area of Triangle is : ",area)
```

**Q.14 WAP to calculate simple interest.**

**Ans:**

```
p=int(input("Enter Principal : "))
r=int(input("Enter Rates : "))
t=int(input("Enter Time : "))
si=(p*r*t)/100
print("Simple Interest is :",si)
```

**Q.15 WAP to read a number in n and prints  $n^2$ ,  $n^3$ ,  $n^4$**

```
Ans: n=int(input("Enter value of n : "))
print("n^2 : ",n*n)
print("n^3 : ",n*n*n)
print("n^4 : ",n*n*n*n)
```

## Chapter – 4: DATA HANDLING

### Very Short answer Type Questions

**Q.1 Identify the data types of the following values given bellow –**

**3, 3j, 13.0, '12', "14", 2+0j, 19, [1,2,3],(3,4,5)**

**Ans:** 3 – int      3j – complex      13.0 – float      '12' – string      "14" – string  
2+0j – complex      19 – int      [1,2,3] – list      (3,4,5) – tuple

**Q.2 What will be the output of the following**

**(a)12/4    (b)14//14    (c)14%4    (d) 14.0/4    (e) 14.0//4    (f)14.0%4**

**Ans:** (a) 3.0      (b) 1      (c) 2      (d) 3.5      (e) 3.0      (f) 2.0

**Q.3 What will be the output of the following ?**

```
print(17//4)
print(17/4)
print(len(str(17//4)))
print(len(str(17/4)))
```

**Ans:** 4  
4.25  
1  
4

**Q.4 What will be the output of the following ?**

**(a) bool(0)                      (b) bool('0')                      (c) bool(int('0'))**  
**(d) bool(str(0.0))              (e) bool(0j)                      (f) bool(0.0)**

**Ans:** (a) False                      (b) True                      (c) False  
(d) True                      (e) False                      (f) False

**Q.5 What will be the output of the following ?**

**(a)87//5    (b)(87//5.0) == (87//5)    (c) 87//5.0    (d) 17%5.0**

**Ans:** (a) 17      (b) True      (c) 17.0      (d) 2.0

**Q.6 int('a') produces error. Why?**

**Ans:** This is because 'a' is an invalid literal for int() with base 10.

**Q.7 Write following expressions in Python.**

**(a)  $\frac{1}{3}b^2h$       (b)  $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$       (c)  $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$       (d)  $a^n \times a^m = a^{n+m}$**

**Ans:** (a) (b\*b\*h)/3  
(b) d=math.sqrt(pow(x2-x1,2)+pow(y2-y1,2))  
(c) x1=(-b) + math.sqrt((b\*b)-(4\*a\*c))/(2\*a)  
x2=(-b) - math.sqrt((b\*b)-(4\*a\*c))/(2\*a)  
(d) pow(a,n) \* pow(a,m) = pow(a,m+n)

---

### Short Answer Type Questions

**Q.1 What are data types? What are Python's built-in core data types?**

**Ans:** Every value in Python has a datatype. Since everything is an object in Python programming, data types are actually classes and variables are instance (object) of these classes.

There are various data types in Python. Some of the important types are listed below.

(i) Numbers (ii) String (iii) List (iv) Tuple (v) Dictionary

**Q.2 Which data types of Python handle Numbers?**

**Ans:** It is cleared by name that Number data types are used to store numeric value in Python. The Numbers in Python have following core data types:

- (i) Integers
  - a. Integers (signed)
  - b. Booleans
- (ii) Floating-Point Numbers
- (iii) Complex Numbers

**Q.3 Why is Boolean considered a subtype of Integers?**

**Ans:** Because Boolean Values *False* and *True* behave like the values 0 and 1, respectively. So Boolean type is a subtype of plain integers.

**Q.4 What do you understand by term 'immutable'?**

**Ans:** Immutable types are those data types that can never change their value in place. In Python the following types are immutable:

- (i) integers
- (ii) floating-point numbers
- (iii) Booleans
- (iv) Strings
- (v) Tuples

**Q.5 What will be the output of the following code? Why?**

**(a) 13 or len(13) (b) len(13) or 13**

**Ans:** (a) 13 (b) TypeError: object of type 'int' has no len().

**Q.6 What are mutable and immutable types in Python? List both of them.**

**Ans: Mutable types** means those data types whose values can be changed at the time of execution.

They are as follows:

- Lists
- Dictionaries
- Sets

Immutable types are those data types that can never change their value in place. In Python the following types are immutable:

- integers
- floating-point numbers
- Booleans
- Strings
- Tuples

**Q.7 What are augmented assignment operators? How are they useful?**

**Ans:** An augmented assignment is generally used to replace a statement where an operator takes a variable as one of its arguments and then assigns the result back to the same variable. A simple example is  $x += 1$  which is expanded to  $x = x + (1)$ . Similar constructions are often available for various binary operators. They are helpful in making the source code small.

**Skill Based Questions****Q.1 WAP to calculate compound simple interest after taking the principle, rate and time.**

**Ans:** `#Compound Interest`  
`p=int(input("Enter the Principal"))`  
`r=int(input("Enter the Interest Rate"))`  
`t=int(input("Enter the Tenure"))`  
`temp=1+r/100`  
`f=1`  
`for i in range(1,t+1):`  
`f=f*temp`  
`Amount=p*f`  
`interest=Amount-p`  
`print("The interest on ",p," with rate ",r," is ",interest)`

**Q.2 WAP to check the given year is leap year or not.**

**Ans:** `year = int(input("Enter year"))`  
`if (year % 4) == 0:`  
`if (year % 100) == 0:`  
`if (year % 400) == 0:`  
`print("{0} is a leap year".format(year))`  
`else:`  
`print("{0} is not a leap year".format(year))`  
`else:`  
`print("{0} is a leap year".format(year))`  
`else:`  
`print("{0} is not a leap year".format(year))`

**Q.3 WAP to take two numbers and check that the first number is fully divisible by second number or not.**

**Ans:** `a=int(input("Enter First Number : "))`  
`b=int(input("Enter Second Number : "))`  
`if a%b==0:`  
`print(a, " is fully divisible by ",b)`  
`else:`  
`print(a, " is not fully divisible by ",b)`

**Q.4 What will be the output of the following?**

```
a=5-4-3
b=3**2**3
print(a)
print(b)
```

**Ans:** -2

6561

**Q.5 What will be the output of the following?**

```
x,y=4,8
z=x/y*y
print(z)
```

**Ans:** 4.0**Q.6 WAP to take value of x,y,z from the user and calculate the equation  $4x^4 + 3y^3 + 9z^2 + 6\pi$** 

```
Ans: x=int(input("Enter x : "))
y=int(input("Enter y : "))
z=int(input("Enter z : "))
f=4*pow(x,4)+3*pow(y,3)+9*pow(z,2)+6*3.14
print("The Answer is : ",f)
```

**Q.7 WAP to take the temperatures of all 7 days of the week and displays the average temperature of that week.**

```
Ans: d1=int(input("Temperature of day 1 : "))
d2=int(input("Temperature of day 2 : "))
d3=int(input("Temperature of day 3 : "))
d4=int(input("Temperature of day 4 : "))
d5=int(input("Temperature of day 5 : "))
d6=int(input("Temperature of day 6 : "))
d7=int(input("Temperature of day 7 : "))
avg=(d1+d2+d3+d4+d5+d6+d7)/7
print("The average temp is : ",avg)
```

**Chapter – 5: CONDITIONAL AND ITERATIVE STATEMENTS****Short Answer Type Questions****Q.1 What a range() function does? Give an example.**

**Ans:** The range() function returns a sequence of numbers, starting from 0 by default, and increments by 1 (by default), and ends at a specified number. its syntax is **range(start, stop, step)** e.g.

```
x = range(3, 6)
```

```
for n in x:
```

```
    print(n)
```

```
#This code will print 3 4 5
```

```
x = range(1, 10,2)
```

```
for n in x:
```

```
    print(n)
```

```
#This code will print 1 3 5 7 9
```

**Q.2 What are loops in Python? How many types of loop are there in Python?**

**Ans:** Loops are iteration constructs in Python. Iteration means repetition of a set of statements depending upon a condition test. Loops has three basic elements within it to repeat the statements –

- Initialization (Start)
- Check Condition (Stop)
- Updation (Step)

Python provide two types of loop

- Conditional Loop **while**( Condition based loop)
- Counting loop **for** (loop for a given number of times).

**Q.3 What is the syntax of if-elif statement in Python?**

**Ans:** The syntax of if-elif statement in python is as follows:

```
If condition1:
```

```
    #code-block of statements when condition1 is true
```

```
elif condion2:
```

```
    #code-block of statements when condition2 is true
```

```

elif condition3:
    #code-block of statements when condition3 is true
    .
    .
    .
else:
    #code-block of statements when all above conditions are false.

```

#### Q.4 What are jump statements in Python? Name jump statements with example.

**Ans:** Python offers two jump statements to be used with in loops to jump out of loop-iterations. These are **break** and **continue** statements.

```

#program to stop loop when 5 encounters
n=int(input("Enter limit : "))
i=0
while i<=n:
    i+=1
    if i==5:
        break
    print(i)

```

```

#program to print upto n excluding 5
n=int(input("Enter limit : "))
i=0
while i<=n:
    i+=1
    if i==5:
        continue
    print(i)

```

#### Q.5 Rewrite the following code fragment using for loop.

```

i=100
while(i>0):
    print(i)
    i-=3

```

**Ans:**

```

for i in range(100,0,-3):
    print(i)

```

#### Q.6 What is the error in following code. Rewrite the correct code.

```

weather='raining'
if weather='sunny':
    print('wear sunblock')
elif weather='snow':
    print('Going skiing')
else:
    print("None of the above")

```

**Correct Code:**

```

weather='raining'
if weather=='sunny':
    print('wear sunblock')
elif weather=='snow':
    print('Going skiing')
else:
    print("None of the above")

```

#### Q.7 Rewrite the following code fragment using while loop.

```

for i in range(1,10):
    if i%3==0:
        print(i)

```

**Ans:**

```

i=1
while i<10:
    if i%3 ==0:
        print(i)
    i+=1

```

#### Skill Based Questions

##### Q.1 WAP that searches for prime numbers from 15 through 25.

**Ans:**

```

for a in range(15,25):
    k=0
    for i in range(2,a//2+1):
        if(a%i==0):
            k=k+1
    if(k==0):
        print(a)

```

**Q.2 WAP to test if given number is prime or not.**

```

Ans: a=int(input("Enter number: "))
      k=0
      for i in range(2,a//2+1):
          if(a%i==0):
              k=k+1
      if(k<=0):
          print("Number is prime")
      else:
          print("Number isn't prime")

```

**Q.3 WAP to compute the result when two numbers and one operator is given by user.**

```

Ans: a=int(input("Enter 1st number: "))
      b=int(input("Enter 2nd number: "))
      c=input("Enter the Operation +,-,/,*:")
      print("The result is : ", end='')
      if c=='+' :
          print(a+b)
      elif c=='-' :
          print(a-b)
      elif c=='/' :
          print(a/b)
      elif c=='*' :
          print(a*b)
      else:
          print("Eoor : Wrog operator entered")

```

**Q.4 WAP to calculate the roots of a given quadratic equation.**

```

Ans: import math
      a=int(input("Enter a "))
      b=int(input("Enter b "))
      c=int(input("Enter c "))
      d=(b*b)-(4*a*c)
      if d>=0:
          print("roots are : ")
          x1=-b+math.sqrt(d)/(2*a)
          x2=-b-math.sqrt(d)/(2*a)
          print(" x1 = =",x1)
          print(" x2 = =",x2)
      else:
          print("roots are imaginary.")

```

**Q.5 WAP to input a digit and print it in words.**

```

Ans: n=int(input("Enter the Digit from 0 to 9: "))
      print("Entered Digit is : ",end='')
      if n==0:
          print("Zero")
      elif n==1:
          print("One")
      elif n==2:
          print("Two")
      elif n==3:
          print("Three")
      elif n==4:
          print("Four")
      elif n==5:
          print("Five")
      elif n==6:
          print("Six")
      elif n==7:
          print("Seven")
      elif n==8:
          print("Eight")
      elif n==9:
          print("Nine")
      else:
          print("Not a digit")

```



**Q.6 WAP to check whether square root of a given number is prime or not.**

```

Ans: import math
n=int(input("Enter a number"))
m=int(math.sqrt(n))
k=0
for i in range(2,m//2+1):
    if(m%i==0):
        k=k+1
if(k==0):
    print(m," which is square root of ",n," is Prime number.")
else:
    print(m," which is square root of ",n," is not Prime number.")

```

**Q.7 WAP to print first n odd numbers in descending order.**

```

Ans: n=int(input("Enter the Limit "))
if n%2==0:
    for i in range(n-1,0,-2):
        print(i)
else:
    for i in range(n,0,-2):
        print(i)

```

**Q.8 WAP to print the following series -**

(i) 1 4 7 10 .....40

(ii) 1 -4 7 -10 ..... -40

```

Ans: (i) n=int(input("Enter the Limit "))
for i in range(1,n+1,3):
    print(i, end=' ')

```

```

(ii) n=int(input("Enter the Limit "))
j=0
for i in range(1,n+1,3):
    print(pow(-1,j)*i, end=' ')
    j+=1

```

**Q.9 WAP to find the average of the list of the numbers entered through keyboard.**

```

Ans: n=int(input("Enter the Limit "))
s=0
for i in range(1,n+1):
    print("Enter ",i,end='')
    a=int(input("th number : "))
    s=s+a
avg=s/n
print("The sum of entered numbers : ",s)
print("The Average of entered numbers : ",avg)

```

**Q.10 WAP to find the largest number from the list of the numbers entered through keyboard.**

```

Ans: n=int(input("Enter the Limit less than 999999999999 "))
small=999999999999
for i in range(1,n+1):
    print("Enter ",i,end='')
    a=int(input("th number : "))
    if a<small:
        small=a
print("Smallest no. is : ",small)

```

**Q.11 WAP to find the 2<sup>nd</sup> largest number from the list of the numbers entered through keyboard. (This program is from List Chapter)**

```

Ans: a=[]
n=int(input("Enter number of elements:"))
for i in range(1,n+1):
    b=int(input("Enter element:"))
    a.append(b)
a.sort()
print("Second largest element is:",a[n-2])

```

**Q.12 WAP to find the sum of n natural numbers.**

```
Ans: n=int(input("Enter the Limit : "))
s=0
for i in range(1,n+1):
    s=s+i
print("The sum is : ",s)
```

**Q.13 WAP to find the sum of first n even numbers.**

```
Ans: n=int(input("Enter the Limit : "))
s=0
for i in range(0,n+1,2):
    s=s+i
print("The sum is : ",s)
```

**Q.14 WAP to find the sum of first n odd numbers.**

```
Ans: n=int(input("Enter the Limit : "))
s=0
for i in range(1,n+1,2):
    s=s+i
print("The sum is : ",s)
```

**Q.15 WAP to print the following pattern**

(a)	*	(b)	*	(c)	A	(d)	0
	**		**		A B		2 2
	***		***		A B C		4 4 4
	****		****		A B C D		8 8 8 8
	*****		*****		A B C D E		

```
Ans: (a) n=int(input("Enter the Limit : "))
for i in range(1,n+1):
    for j in range(1,i+1):
        print("*",end='')
    print("")
```

```
(b) n=int(input("Enter the Limit : "))
for i in range(1,n+1):
    for k in range(n-i,0,-1):
        print(' ',end='')
    for j in range(1,i+1):
        print("*",end='')
    print("")
```

```
(c) s=input("Enter the String : ")
n=len(s)
for i in range(0,n):
    for j in range(0,i):
        print(s[j],end='')
    print("")
```

```
(d) n=int(input("Enter the Limit : "))
    for i in range(0,n+1,2):
        for j in range(0,i+1,2):
            print(i,end='')
        print("")
```

## Chapter – 5: STRING MANIPULATION

### Very Short answer Type Questions

**Q.1** which of the following is not a Python legal string operation?

- (a) 'abc'+ 'abc'                      (b) `abc`\*3                      (c) 'abc' + 3                      (d) 'abc'.lower()

**Ans:** (c) 'abc' + 3

**Q.2** Out of the following operators, which ones can be used with strings?

=, -, \*, /, //, %, >, <>, in, not in, <=

**Ans:** /, // and %

**Q.3** From the string S = "CARPE DIEM". Which ranges return "DIE" and "CAR"?

**Ans:** S[6:9] for "DIE" and S[0:3] for "CAR"

**Q.4** Given a string S = "CARPE DIEM". If n is length/2 then what would following return?

- (a) S[:n]    (b) S[n:]    (c) S[n:n]    (d) S[1:n]    (e) S[n:length-1]

**Ans:** (a) "CARPE" (b) "DIEM" (c) '' (d) "ARPE" (e) "DIE"

**Q.5** What would following expression return?

- (a) "Hello World".upper().lower()                      (b) "Hello World".lower().upper()  
 (c) "Hello World".find("Wor",1,6)                      (d) "Hello World".find("Wor")  
 (e) "Hello World".find("wor")                      (f) "Hello World".isalpha()  
 (g) "Hello World".isalnum()                      (h) "Hello World".isdigit()  
 (i) "123FGH".isdigit()

**Ans:** (a) 'hello world'

(b) 'HELLO WORLD'

(c) -1

(d) 6

(e) -1

(f) False

(g) False

(h) False

(i) False

### Short Answer Type Questions

**Q.1** What is a string slice? How is it useful?

**Ans:** String Slice is a part of a string containing some contiguous characters from the string. It is accessed from the string by providing a range in "[ ]" brackets i.e. S [n:m]. Python returns all the characters at indices n, n+1, n+2 . . . m-1 e.g.

'Barabanki'. [4:7] will return 'ban'.

**Q.2** Write a python script that traverses through an input string and prints its characters in different lines – two characters per line.

**Ans:**

```
s=input("Enter a string")
l=len(s)
j=0
for i in range(0,l):
    print(s[i],end='')
    j+=1
    if j%2==0:
        print('')
```

**Q.3 Which functions would you chose to use to remove leading and trailing white spaces from a given string?**

**Ans:** Python String strip() function will remove leading and trailing whitespaces. If you want to remove only leading or trailing spaces, use lstrip() or rstrip() function instead.

**Q.4 Suggest appropriate functions for the following tasks –**

- (a) To check whether the string contains digits.
- (b) To find the occurrence a string within another string.
- (c) To convert the first letter of a string to upper case.
- (d) To convert all the letters of a string to upper case.
- (f) To check whether all the letters of the string are in capital letters.
- (g) to remove all the white spaces from the beginning of a string.

**Ans:** (a) isalnum()                      (b) find()                      (c) capitalize()  
(d) upper()                              (f) isupper()                      (g) lstrip()

**Q.5 Find the errors -**

```
s="PURA VIDA"
Print(s[9] + s[9:15])
```

**Ans:** Here the error is : Sting index out of range.

**Q.6 Find the output – if we give input as "Hello"**

```
S=input("Enter String")
RS=""
for ch in S:
    RS=ch+RS
print(S+RS)
```

```
S=input("Enter String")
RS=""
for ch in S:
    RS=ch*2+RS
print(RS+S)
```

**Ans: output 1**    Enter a stringHello  
HelloH  
HelloeH  
HelloleH  
HellolleH  
HelloolleH

**output2**    Enter a stringHello  
HelloHH  
HelloeeHH  
HellolleeHH  
HellolllleeHH  
HellooollleeHH

### Skill Based Questions

**Q.1 WAP to print following pattern without using any nested loop.**

```
#
# #
# # #
# # # #
# # # # #
```

**Ans:**    n=int(input("Enter Limit"))  
for i in range(1,n+1):  
    print("#"\*i)

**Q.2 WAP to print the number of occurrences of a substring into a line.**

**Ans:**

```
s=input("Enter a String : ")
substr=input("Enter a Sub String : ")
l=len(s)
lsub=len(substr)
start=count=0
end=l
while True:
    position=s.find(substr, start, end)
    if position!=-1:
        count+=1
        start=position+lsub
    else:
        break
    if start>=l:
        break
print("No. of occurances of ",substr, " : ",count)
```

**Q.3 WAP to check the given string is palindrome or not.**

```

Ans: s=input("Enter a string : ")
mid=len(s)//2
rev=-1
for a in range(mid):
    if s[a]==s[rev]:
        rev-=1
    else:
        print(s," is not Palindrome")
        break
else:
    print(s, " is palindrome")

```

**Q.4 WAP that:**

- **Prompt the user for a string**
- **Extract all the digits from the string.**
- **If there are digits**
  - **Sum the collected digits together.**
  - **Printout:**
    - **The original string**
    - **The digits**
    - **The sum of the digits**
- **If there are no digits**
  - **Print the original string**
  - **A message "Has no Digits"**

```

Ans: s=input("Enter a String : ")
sum=0
num=0
if s.isalpha()==False:
    for i in s:
        if i.isdigit()==True:
            num=num*10+int(i)
            sum+=int(i)
    print("Original String : ",s)
    print("Digits : ",num)
    print("Sum of digits is : ",sum)
else:
    print("Original String ",s," has no digit")

```

**Chapter – 7: LIST MANIPULATION****Very Short answer Type Questions****Q.1 What do you understand by mutability?**

**Ans:** Mutable means changeable. In Python, mutable types are those whose values can be changed in place. Only three types are mutable in python – Lists, Dictionaries and Sets.

**Q.2 Start with the list[8,9,10]. Do the following using list functions**

- (a) Set the second entry (index 1) to 17**
- (b) Add 4, 5 and 6 to the end of the list.**

**(c) Remove the first entry from the list.**

**(d) Sort the list.**

**(e) Double the list.**

**(f) Insert 25 at index 3**

- Ans:** (a) `list[1]=17`  
 (b) `list.append(4)`  
       `list.append(5)`  
       `list.append(6)`  
 (c) `list.pop(0)`  
 (d) `list.sort()`  
 (e) `list=list*2`  
 (f) `list.insert(3,25)`

**Q.3 If a is [1, 2, 3], what is the difference (if any) between a\*3 and [a, a, a]?**

**Ans:** `a*3` will produce `[1,2,3,1,2,3,1,2,3]`, means a list of integers and `[a, a, a]` will produce `[[1,2,3],[1,2,3],[1,2,3]]`, means list of lists

**Q.4 If a is [1, 2, 3], is a \*3 equivalent to a + a + a?**

**Ans:** Yes, Both `a*3` and `a+a+a` will produce same result.

```
>>> a=[1, 2, 3]
>>> a*3
[1, 2, 3, 1, 2, 3, 1, 2, 3]
>>> a+a+a
[1, 2, 3, 1, 2, 3, 1, 2, 3]
```

**Q.5 If a is [1, 2, 3], what is the meaning of a [1:1] = 9?**

**Ans:** This will generate an error "TypeError: can only assign an iterable".

**Q.6 If a is [1, 2, 3], what is the meaning of a [1:2] = 4 and a [1:1] = 4?**

**Ans:** These will generate an error "TypeError: can only assign an iterable".

**Q.7 What are list slices?**

**Ans:** List slices are the sub-part of a list extracted out. You can use indexes of the list elements to create *list slices* as per following format. Syntax is as follows –

*Seq=ListName[start:stop]*

**Q.8 Does a slice operator always produce a new list?**

**Ans:** Yes, this will create a new list.

### Short Answer Type Questions

**Q.1 How are lists different from strings when both are sequences?**

**Ans:** Lists are similar to strings in many ways like indexing, slicing, and accessing individual elements but they are different in the sense that

- (i) *Lists are mutable while strings are immutable.*
- (ii) *In consecutive locations, strings store the individual characters while list stores the references of its elements.*
- (iii) *Strings store single type of elements-all characters while lists can store elements belonging to different types.*

**Q.2 What are nested Lists?**

**Ans:** A list can have an element in it, which itself is a list. Such a list is called nested list. e.g.

`L = [1,2,3,4,[5,6,7],8]`

**Q.3 Discuss the utility and significance of Lists.**

**Ans:** The list is a most versatile datatype available in Python which can be written as a list of comma-separated values (items) between square brackets. Important thing about a list is that items in a list need not be of the same type. List is majorly used with dictionaries when there is large number of data.

**Q.4 What is the purpose of the del operator and pop method? Try deleting a slice.**

**Ans:** **del operator** is used to remove an individual item, or to remove all items identified by a slice. It is to be used as per syntax given below –

```
>>>del List[index]
```

```
>>>del List[start:stop]
```

**pop method** is used to remove single element, not list slices. The pop() method removes an individual item and returns it. Its syntax is –

```
>>>a=List.pop() #this will remove last item and deleted item will be assigned to a.
```

```
>>>a=List[10] # this will remove the ite at index 10 and deleted item will be assigned to a.
```

**Q.5 What are list slices?**

**Ans:** List slices, like string slices are the sub part of a list extracted out. Indexes can be used to create list slices as per following format:

```
seq = L[start:stop]
```

**Q.6 What do you understand by true copy of a list? How is it different from shallow copy?**

**Ans:** A **shallow copy** means constructing a new collection object and then populating it with references to the child objects found in the original. In essence, a shallow copy is only *one level deep*. The copying process does not recurse and therefore won't create copies of the child objects themselves.

**True Copy means** you can create a copy of a list using `New_list=My_list`. The assignment just copies the reference to the list, not the actual list, so both `new_list` and `my_list` refer to the same list after the assignment.

**Q.7 Predict the output –**

```
L1=[1,3,5,7,9]
print(L1==L1.reverse())
print(L1)
```

**Ans:** False  
[9, 7, 5, 3, 1]

**Q.8 Predict the output –**

```
List1=[13,18,11,16,13,18,13]
print(List1.index(18))
print(List1.count(18))
List1.append(List1.count(13))
print(List1)
```

**Ans:** 1  
2  
[13, 18, 11, 16, 13, 18, 13, 3]

**Skill Based Questions****Q.1 WAP to find minimum element from a list of elements along with its index in the list.**

**Ans:**

```
L=[2,58,95,999,65,32,15,1,7,45]
print(L)
m=min(L)
print("The Minimum elements is :",m)
print("Index of minimum element is :",L.index(m))
```

This program is for 1 or 2 marks

**Or**

```
lst=eval(input("Enter List : "))
length=len(lst)
min_ele=lst[0]
min_index=0
for i in range(1,length-1):
    if lst[i]<min_ele:
        min_ele=lst[i]
        min_index=i
print("Given list is :",lst)
print("The Minimum element of the list is : ")
print(min_ele," at index ",min_index)
```

This program is for 3 or 4 marks

**output**

```
Enter List : [67,87,56,6,4,89,99,3,2,56]
Given list is : [67, 87, 56, 6, 4, 89, 99, 3, 2, 56]
The Minimum element of the list is :
2 at index 8
```

**Q.2 WAP to calculate mean of a given list of numbers.**

**Ans:**

```
from statistics import mean
L=[2,58,95,999,65,32,15,1,7,45]
m=mean(L)
print("Mean of list is : ",m)
```

This program is for 1 or 2 marks

**OR**

```
lst=eval(input("Enter List : "))
length=len(lst)
mean=sum=0
for i in range(0,length-1):
    sum+=lst[i]
mean=sum/length
print("Given List is : ",lst)
print("The Mean of given list is : ",mean)
```

This program is for 3 or 4 marks

**output**

```
Enter List : [1,2,3,4,5,6,7,8,9]
Given List is : [1, 2, 3, 4, 5, 6, 7, 8, 9]
The Mean of given list is : 4.0
```

**Q.3 WAP to search for an element in a given list of numbers.**

**Ans:**

```
L=[2,58,95,999,65,32,15,1,7,45]
n=int(input("Enter the number to be searched : "))
found=0
for x in L:
    if x==n:
        print("Item found at position : ",L.index(n)+1)
        found=1
if found==0:
    print("Item not found")
```

**Q.4 WAP to count frequency of a given element in a list of numbers.**

**Ans:**

```
L=[2,58,95,999,95,32,15,95,7,45]
n=int(input("Enter the number : "))
print("The frequency of number ",n," is ",L.count(n))
```

This program is for 1 or 2 marks

**Or**

```
lst=eval(input("Enter List : "))
length=len(lst)
element=int(input("Enter the element: "))
count=0
for i in range(0,length-1):
    if element==lst[i]:
        count+=1
if count==0:
    print(element," Not Found")
else:
    print(element," has frequency as ",count," in given list")
```

This program is for 3 or 4 marks

**output**

```
Enter List : [34,45,56,45,34,88,45,99,45,90,45]
Enter the element: 45
45 has frequency as 4 in given list
```

**Q.5 WAP to calculate the sum of integers of the list.**

**Ans:**

```
L=[1,2,"Two",3,4,"four",5]
s=0
for x in L:
    if type(x)==int:
        s=s+x
print("The sum of integers is : ",s)
```

**Q.6 WAP that reverses an array of integers (in place)**

**Ans:**

```
alist=[1,2,3,4,5,6]
end = len(alist)-1
limit = int(end/2) + 1
for i in range(limit):
    alist[i],alist[end] = alist[end],alist[i]
    end = end - 1
print (alist)
```



**Q.7 WAP that creates a third list after adding two lists.**

```

Ans: lst1=eval(input("Enter List1 : "))
      lst2=eval(input("Enter List2 : "))
      lst3=lst1+lst2
      print("Given List are : ", lst1," and ",lst2)
      print("The sum of list are : ",lst3)

```

**Q.8 WAP a program to generate a list of elements of Fibonacci Series.**

```

Ans: n=int(input("Enter the limit :"))
      a=-1
      b=1
      for i in range(1,n+1):
          c=a+b
          print(c,end=' ')
          a=b
          b=c

```

Enter the limit :10  
0 1 1 2 3 5 8 13 21 34

**Chapter – 8: DICTIONARIES****Very Short answer Type Questions****Q.1 Why can't List can be used as keys?**

**Ans:** List is mutable datatype. And Keys of dictionary must be immutable type. This is the region that list cannot be used as keys.

**Q.2 What type of objects can be used as keys in dictionary?**

**Ans:** Any immutable objects can be used as keys in dictionary.

**Q.3 Can you change the order of the dictionaries contents?**

**Ans:** Yes, the order of dictionary contents may be changed.

**Q.4 Can you modify the keys in a dictionary?**

**Ans:** Keys cannot be modified in dictionary while values are mutable in dictionary.

**Q.5 Can you modify the value in a dictionary?**

**Ans:** Yes, Values can be modified in dictionaries.

**Q.6 Is dictionary Mutable? Why?**

**Ans:** Dictionary is mutable type because we can change its values.

**Short Answer Type Questions****Q.1 How are dictionaries different from Lists?**

**Ans:** The dictionary is similar to lists in the sense that it is also a collection of data-items but it is different from lists in the sense that lists are sequential collections(ordered) and dictionaries are non-sequential collections(unordered).

Elements in lists or tuples can be accessed by using indexes. But in dictionaries the values can be obtained using keys. By changing the sequence of key we can shuffle the order of elements of dictionary while this thing is not possible in list.

**Q.2 When are dictionaries more useful than lists?**

**Ans:** Dictionaries can be much more useful than lists when we wanted to store all our friends cell-phone numbers. We could create a list of pairs,(name of friend, phone number), but once this list becomes long enough searching this list for a specific phone number will get-time consuming. Better would be if we could index the list by our friend's name. This is precisely what a dictionary does.

**Q.3 Discuss the utility and significance of Dictionaries.**

**Ans:** Dictionaries can be much more useful than lists when we wanted to store all our friends cell-phone numbers. We could create a list of pairs,(name of friend, phone number), but once this list becomes long enough searching this list for a specific phone number will get-time consuming. Better would be if we could index the list by our friend's name. This is precisely what a dictionary does.

**Q.4 Why is a dictionary termed as an unordered collection of objects?**

**Ans:** But in dictionaries the values can be obtained using keys. By changing the sequence of key we can shuffle the order of elements of dictionary while this thing is not possible in list. In dictionaries there is no index. It uses its keys as index which can be rearranged. That's why a dictionary termed as an unordered collection of objects

**Q.5 How is clear() function different from del <dict> Statement?**

**Ans:** clear() removes all the elements of a dictionary and makes it empty dictionary while **del** statement removes the complete dictionary as an object. After del statement with a dictionary name, that dictionary object no longer exists, not even empty dictionary.

**Q.6 What is the output produced by the following code -**

```
d1={5:[6,7,8],"a":(1,2,3)}      Ans: dict_keys([5, 'a'])
print(d1.keys())                dict_values([[6, 7, 8], (1, 2, 3)])
print(d1.values())
```

**Skill Based Questions**

**Q.1 WAP that repeatedly asks the user to enter product names and prices. Store all of them in a dictionary whose keys are product names and values are prices. And also write a code to search an item from the dictionary.**

**Ans:**

```
dict={}
ch='y'
while ch=='y' or ch=='Y':
    name=input("Enter name of product :")
    price=eval(input("Enter Price of product :"))
    dict[name]=price
    ch=input("Want to add more items (Y/N) : ")
print(dict)
nm=input("Enter the product you want to search :")
for x in dict:
    if x==nm:
        print("Product found and the price of product \"
            ,x,\" is \",dict[x])
```

```
Enter name of product :Pen
Enter Price of product :10
Want to add more items (Y/N) : y
Enter name of product :Pencil
Enter Price of product :5
Want to add more items (Y/N) : y
Enter name of product :Rubber
Enter Price of product :4
Want to add more items (Y/N) : y
Enter name of product :Cutter
Enter Price of product :4
Want to add more items (Y/N) : n
{'Pen': 10, 'Pencil': 5, 'Rubber': 4, 'Cutter': 4}
Enter the product you want to search :Rubber
Product found and the price of product Rubber is 4
```

**Q.2 WAP to create a dictionary named year whose keys are month names and values are their corresponding number of days.**

**Ans:**

```
dict={}
ch='y'
while ch=='y' or ch=='Y':
    month=input("Enter name of month :")
    days=eval(input("Enter no. of days of month :"))
    dict[month]=days
    ch=input("Want to add more months (Y/N) : ")
print(dict)
```

## Chapter – 9: INTRODUCING PYTHON MODULES

### Very Short answer Type Questions

**Q.1 Write a python code to import math module.**

**Ans:** import math

**Q.2 How can the function sqrt() be used in Python?**

**Ans:** from math import sqrt (importing a single object from a module)

**Q.3 find the error in the following code –**

(a) `def main()  
print("Hello")`

**Ans:** `def main():  
print("Hello")`

(b) `func2():  
print(2+3)`

**Ans:** `def func2():  
print(2+3)`

### Short Answer Type Questions

**Q.1 What is the significance of modules?**

**Ans:** The process of breaking a bigger unit into individual components (modules) is called modularity. Modularity reduces complexity to some degree.

**Q.2 What are docstrings? What is their significance? Give example to support your answer.**

**Ans:** docstrings are triple quoted comments which are useful for documentation purposes. A docstring should be the first string stored inside a module/function-body/class.

For ex-

`""" Conversion function between Fahrenheit and centigrade """`

**Q.3 What is dot notation of referring to objects inside a module?**

**Ans:** After importing a module, to access one of the functions, we have to specify the name of the module and name of the function, separated by a dot. This format is called dot notation.

**Q.4 Why should the from <module> import <objct> statement be avoided to import objects?**

**Ans:** If your program already has a variable with the same name as the one imported via module, then the variable in your program will hide imported member with same name because there cannot be two variables with the same name in one namespace.

**Q.5 What do you understand by standard library of Python?**

**Ans:** Python library is a collection of functions and methods that allows you to perform many actions.

**Q.6 Explain the difference between *import<module>* and *from<module> import statements, with examples.***

**Ans:** import <module> will import the whole module in the program whereas from <module> import will import only asked functions and variables from the module.

For ex-

Import time

From math import sqrt, pow

**Q.7 What is a function? How is it useful?**

**Ans:** A function is a named block in a program which has some certain working. Whenever you call this it performs all its working. It reduces the complexity and reduces the labour of writing same code again and again.

**Q.8 What is an argument? Give an example of one.**

**Ans:** An argument is the requirement of a function to perform its working. For ex- `pow(x,y)`, here x and y are the arguments which are required to pass to `pow()` for its working.

**Skill Based Questions**

Q.1 Create a module with functions for future value, present value and compound interest.

**Chapter – 10: INTRODUCING PYTHON PANDAS****Very Short answer Type Questions**

**Q.1 Write the shape of the following ndarrays:**

(a) 

2	4	6	8
---	---	---	---

(b) 

2
4
6

(c) 

1	2	3	4
5	1	6	7

**Ans:** (a) (4,) (b) (3,) (c) (2,4)

**Q.2 NumPy stands for .....**

**Ans:** Numerical Python.

**Q.3 How to install Python Pandas? Write command.**

**Ans:** `pip install pandas`.

**Q.4 NumPy is installed when you install Pandas. (True/False).**

**Ans:** True.

**Short Answer Type Questions**

**Q.1 What is Pandas library of Python? What is its significance?**

**Ans:** Pandas library of Python is for data analysis. It makes available various tools for data analysis and makes it simple and easy process.

**Q.2 What do you understand by axes in a NumPy array? Define axes for a 2D ndarray?**

**Ans:** Axes of an array describes the order of indexing into the array. For ex- `axes=0` refers to the first index coordinate, `axis=1` the second etc.

**Q.3 How is Series data structure different from a DataFrame data structure?**

**Ans:** A Series is a one-dimensional object that can hold any data type such as integer, float and strings. It has only one axis(0).

A DataFrame is a two dimensional object that can have columns with potential different types. It has two axis(axis 0 and 1).

**Q.4 Name some common data structures of Python's Pandas library?**

**Ans:** Numpy Array, Series, DataFrame.

**Q.5 If a Python list is having 7 integers and a numpy array is also having 7 integers, then how are these two data structure similar or different from one another?**

**Ans:**

1. Numpy array will be occupying less space than list.
2. Vectorized operation is possible with numpy array but not with list.
3. The process of accessing elements is same in both.

**Q.6 Given a list L = [3,4,5] and an ndarray N having elements 3,4,5, what will be the result produced by:**

- (a)  $L * 3$                       (b)  $N * 3$                       (c)  $L + L$                       (d)  $N + N$

Ans: (a)[3,4,5,3,4,5,3,4,5]                      (b) ([9,12,15])                      (c) [3,4,5,3,4,5]                      (d) ([6,8,10])

**Q.7 Write code to create an ndarray having six zeros in it. Write statements to change 3<sup>rd</sup> and 5<sup>th</sup> elements of this ndarray to 15 and 25 respectively.**

```
Ans: >>> import pandas as pd
>>> import numpy as np
>>> ar=np.zeros([2,3],dtype=np.int64)
>>> ar
array([[0, 0, 0],
       [0, 0, 0]], dtype=int64)
>>> ar[0,2]=15
>>> ar[1,1]=25
>>> ar
array([[ 0,  0, 15],
       [ 0, 25,  0]], dtype=int64)
>>> |
```

**Q.8 How is a series object different from and similar to ndarrays? Support your answer with example.**

**Ans:** Series object is a one dimensional object that contains array like values and associated indexes of those values. ndarrays are also similar to series objects but there are some differences that makes them distinctively different types. which are-

1. in ndarrays, you can perform vectorized operations only if the shapes of two ndarrays match, otherwise it returns an error. but with series objects, the data of two series object is aligned as per matching indexes for non-matching indexes, NaN is returned.
2. in ndarrays, indexes are always numeric starting from 0 onwards, but series object can have any type of indexes including numbers, letters, labels, strings etc.

```
>>> l=[1,2,3,4]
>>> l1=[2,3,4,5,6]
>>> a1=np.array(l)
>>> a1
array([1, 2, 3, 4])
>>> a2=np.array(l1)
>>> a2
array([2, 3, 4, 5, 6])
>>> a1+a2
Traceback (most recent call last):
  File "<pyshell#13>", line 1, in <module>
    a1+a2
ValueError: operands could not be broadcast together with shapes (4,) (5,)
```

```
>>> s1
0    1
1    2
2    3
3    4
dtype: int32
>>> s2=pd.Series(a2)
>>> s2
0    2
1    3
2    4
3    5
4    6
dtype: int32
>>> s1+s2
0    3.0
1    5.0
2    7.0
3    9.0
4    NaN
dtype: float64
```

**Q.9 Write a python code to create a series object Temp1 that stores temperatures of seven days in it. Take any random seven temperatures.**

```
>>> ar=np.linspace(22,42,7)
>>> ar
array([22.000000, 25.33333333, 28.66666667, 32.000000, 35.33333333,
       38.66666667, 42.000000])
```

```
>>> temp1=pd.Series(ar)
>>> temp1
0    22.000000
1    25.333333
2    28.666667
3    32.000000
4    35.333333
5    38.666667
6    42.000000
dtype: float64
```

**Q.10** Write a python code to create a series object *Temp1* that stores temperatures of seven days of week. Its indexes should be 'Sunday', 'Monday', . . . 'Saturday'.

**Ans:**

```

>>> temp1=pd.Series(ar,index=['Sunday','Monday','Tuesday','wednesday','Thursda
y','Friday','Saurday'])
>>> temp1
Sunday          22.000000
Monday          25.333333
Tuesday         28.666667
wednesday       32.000000
Thursday        35.333333
Friday          38.666667
Saurday         42.000000
dtype: float64

```

**Q.11** Write commands to print following details of series object *seal*.

(a) If the series is empty.

(b) Indexes of the series.

(c) The data type of underlying data

(d) If the series stores any NaN value.

**Ans:**

(a) 

```
>>> ser=pd.Series()
>>> ser
Series([], dtype: float64)
```

(c) 

```
>>> temp1.dtype
dtype('float64')
```

(b) 

```
>>> temp1.index
Index(['Sunday', 'Monday', 'Tuesday', 'wednesday', 'Thursday', 'Friday',
      'Saurday'],
      dtype='object')
```

(d) 

```
>>> s=s1+s2
>>> s
0    3.0
1    5.0
2    7.0
3    9.0
4    NaN
dtype: float64
>>> s.hasnans
True
```

### Skill Based Questions

**Q.1** given the following series objects:

**S1**

0	3
1	5
2	6
4	10
5	12

**S2**

0	12
2	10
3	15
4	20
6	27

(a) what will be the result of S1 + S2?

(b) What will be the result of S1 - S2?

**Ans: (a)**

```

>>> s1
0    3
1    5
2    6
4   10
5   12
dtype: int64
>>> s2=pd.Series([12,10,15,20,27], index=[0,2,3,4,6])
>>> s2
0    12
2    10
3    15
4    20
6    27
dtype: int64
>>> s1+s2
0    15.0
1    NaN
2    16.0
3    NaN
4    30.0
5    NaN

```

**(b)**

```

>>> s1-s2
0   -9.0
1    NaN
2   -4.0
3    NaN
4  -10.0
5    NaN
6    NaN
dtype: float64
>>>

```

**Q.2 Consider the following Series object namely S:**

0	0.430271
1	0.617328
2	-0.265421
3	-0.836113

dtype: float64

What will be returned by following statements?

(a)  $S * 100$       (b)  $S > 0$       (c)  $S1 = \text{pd.Series}(S)$       (d)  $S2 = \text{pd.Series}(S1) + 3$

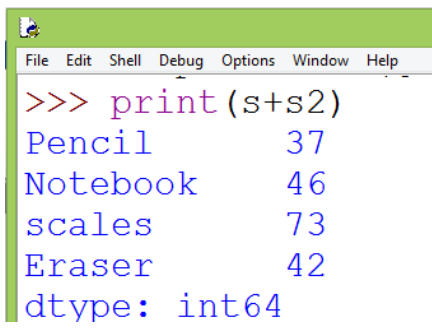
What will be the values of Series objects S1 and S2 created above ?

<p><b>(a)</b></p> <pre>&gt;&gt;&gt; s=pd.Series([.430271,0.617328,-0.265421,-0.836113]) &gt;&gt;&gt; s 0    0.430271 1    0.617328 2   -0.265421 3   -0.836113 dtype: float64 &gt;&gt;&gt; s*100 0    43.0271 1    61.7328 2   -26.5421 3   -83.6113 dtype: float64 &gt;&gt;&gt;</pre>	<p><b>(b)</b></p> <pre>&gt;&gt;&gt; s&gt;0 0     True 1     True 2    False 3    False dtype: bool &gt;&gt;&gt;</pre>
<p><b>(c)</b></p> <pre>&gt;&gt;&gt; s1=pd.Series(s) &gt;&gt;&gt; s1 0    0.430271 1    0.617328 2   -0.265421 3   -0.836113 dtype: float64 &gt;&gt;&gt;</pre>	<p><b>(d)</b></p> <pre>&gt;&gt;&gt; s2=pd.Series(s1)+3 &gt;&gt;&gt; s2 0    3.430271 1    3.617328 2    2.734579 3    2.163887 dtype: float64 &gt;&gt;&gt;</pre>

**Q.3 What will be the output produced by: (suppose pandas has been imported as pd)**

```
Stationary=['pencils', 'notebooks', 'scales', 'erasers']
s=pd.Series([20,33,52,10], index=Stationary)
s2=pd.Series([17,13,31,32], index=Stationary)
print(s+s2)
s=s+s2
print(s+s2)
```

**Ans:**

 <pre>&gt;&gt;&gt; print(s+s2) Pencil      37 Notebook    46 scales      73 Eraser      42 dtype: int64</pre>	<pre>&gt;&gt;&gt; s=s+s2 &gt;&gt;&gt; print(s+s2) Pencil      54 Notebook    59 scales      94 Eraser      74 dtype: int64 &gt;&gt;&gt;</pre>
--	---

**Q.4** What will be the output produced by following code, considering the Series object S given in above question.

(a) `print(s[1:1])`

(b) `print(s[0:1])`

(c) `print(s[0:2])`

(d) `s[0:2]=12`

(e) `print(s.index)`

`Print(s)`

`print(s.values)`

**Ans:**

```
dtype: int64
>>> print(s[1:1])
Series([], dtype: int64)
>>> print(s[0:1])
Pencil    37
dtype: int64
>>> print(s[0:2])
Pencil    37
Notebook  46
dtype: int64
>>> s[0:2]=12
>>> print(s)
Pencil    12
Notebook  12
scales    73
Eraser    42
dtype: int64
>>> print(s.index)
Index(['Pencil', 'Notebook', 'scales', 'Eraser'], dtype='object')
>>> print(s.values)
[12 12 73 42]
```

**Q.5** Find the error and correct:

```
data=np.array(['a','b','c','d','e','f'])
s=pd.Series(data,index=[100,101,102,103,104,105])
print(s[102,103,104])
```

**Ans:**

```
data=np.array(['a','b','c','d','e','f'])
s=pd.Series(data,index=[100,101,102,103,104,105])
print(s[2:5])
```

**Q.6** if Ser is a Series type object having 30 values, then how are statements (a), (b) and (c), (d) similar and different?

(a) `print(Ser.head())`

(b) `print(Ser.head(8))`

(c) `print(Ser.tail())`

(d) `print(Ser.tail(11))`

**Ans:** (a) will print top 5 values (b) will print top 8 values  
(c) will print 5 bottom values (d) will print 11 bottom values

## CHAPTER– 11: PYTHON PANDAS II-DataFrames and Other Operations

### Short Answer Type Questions

**Q.1** How does dataframe object specify indexes to its data rows?

Ans-It has two indexes- a row index(axis=0) and a column index(axis=1)

**Q.2** Write code statements to list the following, from a dataframe namely sales.

(a) List only columns 'item' and 'Revenue'.

(b) List rows from 3 to 7.

(c) List the value of cell in 5<sup>th</sup> row and, 'item' column.

Ans- (a)`sales.loc ['item', 'Revenue']`

(b)`sales.iloc[3:7]`



---

```
(c)sales.item[5]
```

**Q.3 How would you add a new column namely 'val' to a dataframe df that has 10 rows in it and has columns as 'Item', 'Qty', 'Price'? you can choose to put any values of your choice.**

```
Ans-df['val']=[10,11,12,13,14,15,16,17,18,19]
```

**Q.4 Write code to print information about a series object.**

```
Ans- <series object>.index
      <series object>.values
```

**Q.5 Write code to get summary details about a dataframe object for all its columns including numeric and string columns.**

```
Ans-<DataFrame object>.index
      <DataFrame object>.columns
      <DataFrame object>.axes
      <DataFrame object>.dtype
```

**Q.6 The head() and tail() extract rows or columns from a dataframe. Explain.**

Ans-head() and tail() functions retrieve 5 top and 5 bottom rows respectively of a DataFrame object. The number of rows to be displayed can be changed by specifying a number as parameter. For ex-

```
Df.head()           // will display top 5 rows
Df.tail()           // will display bottom 5 rows
Df.head(3)          // will display top 3 rows
Df.tail(7)          // will display bottom 7 rows
```

**Q.7 Why does Python change the datatype of a column as soon as it stores an empty value (NaN) even though it has all over values stored as integer?**

Ans: Because in Python, integer types cannot store NaN values.

**Q.8 What is a DataFrame object? How it is created in Python?**

Ans: A DataFrame object is a two-dimensional labeled array which is actually an ordered collection of columns where columns may store different types of data.

A dataframe object can be created by following ways-

1. From a 2-D Dictionary
2. From a 2-D ndarray
3. From a 2-D Dictionary with values as Series object
4. From an another DataFrame object

#### Skill Based Questions

**Q.1 Given a dataframe df as shown bellow**

	A	B	C
0	15	17	19
1	16	18	20
2	20	21	22

**What will be the result of the following code segment.**

(a) `df['c'] = np.NaN`      (b) `df['c'] = [2,5]`      (c) `df['c'] = [12,15,27]`

Ans: (a)

```
>>> dict={'A':[15,16,20], 'B':[17,18,21], 'C':[19,20,22]}
>>> df=pd.DataFrame(dict)
>>> df
   A  B  C
0  15 17 19
1  16 18 20
2  20 21 22
>>> df['C']=np.NaN
>>> df
   A  B  C
0  15 17 NaN
1  16 18 NaN
2  20 21 NaN
```

(b) Will raise an error-Length of values does not match length of index

(c)

```
>>> df['C']=[12,15,27]
>>> df
   A  B  C
0  15 17 12
1  16 18 15
2  20 21 27
>>>
```

**Q.2** Write code statements for a dataframe df for the following -

- Delete an existing column from it.
- Delete rows from 3 to 6 from it.
- Check if the dataframe has any missing values.
- fill all missing values with 999 in it.

**Ans:** (a) `del df[<column name>]`

(b) `df.drop([3,4,5,6])`

(c) `df.isnull()`

(d) `Df.fillna(999)`

**Q.3** Predict the output of the following code -

```
my_di={"name":["Hari","Prakash","Suresh"],
       "age":np.array([10,15,20]),
       "weight":(75,123,239),
       "height":[4.5,5,6.1],
       "siblings":1,
       "gender":"M"}
df=pd.DataFrame(my_di)
print(df)
```

Ans:

```
   name  Age  weight  height  siblings  gender
0  Hari   10     75     4.5         1        M
1  Prakash  15    123     5.0         1        M
2  Suresh  23    239     6.1         1        M
>>>
```

**Q.4** Predict the output of the following code -

```
my_di={"name":["Hari","Prakash","Suresh"],
       "age":np.array([10,15,20]),
       "weight":(75,123,239),
       "height":[4.5,5,6.1],
       "siblings":1,
       "gender":"M"}
df2=pd.DataFrame(my_di,index=my_di["name"])
print(df2)
```

**Ans:**

	name	Age	weight	height	siblings	gender
	Hari	10	75	4.5	1	M
	Prakash	15	123	5.0	1	M
	Suresh	23	239	6.1	1	M

**Q.5 Predict the output of the following code –**

```
my_di={"name":["Hari","Prakash","Suresh"],
      "age":np.array([10,15,20]),
      "weight":(75,123,239),
      "height":[4.5,5,6.1],
      "siblings":1,
      "gender":"M"}
df2=pd.DataFrame(my_di,index=my_di["name"])
print(df2["weight"])
print(df2.weight['Prakash'])
```

**Ans:**

```
Hari      75
Prakash   123
Suresh    239
Name: weight, dtype: int64
123
>>> |
```

**Q.6 What will be the output produced by the following code:**

```
df=pd.DataFrame([('The Journey','J.R.R.T. '),
                ('Big Day','Leela D'),
                ('Hashtag World','Ani M')])
print(df)
df.columns=['Title','Author']
print(df|)
```

**Ans:**

```
          0      1
0  The Journey  J.R.R.T
1    Big Day   Leela D
2  Hashtag Worls  Ani M
   Title      Author
0  The Journey  J.R.R.T
1    Big Day   Leela D
2  Hashtag Worls  Ani M
>>> |
```

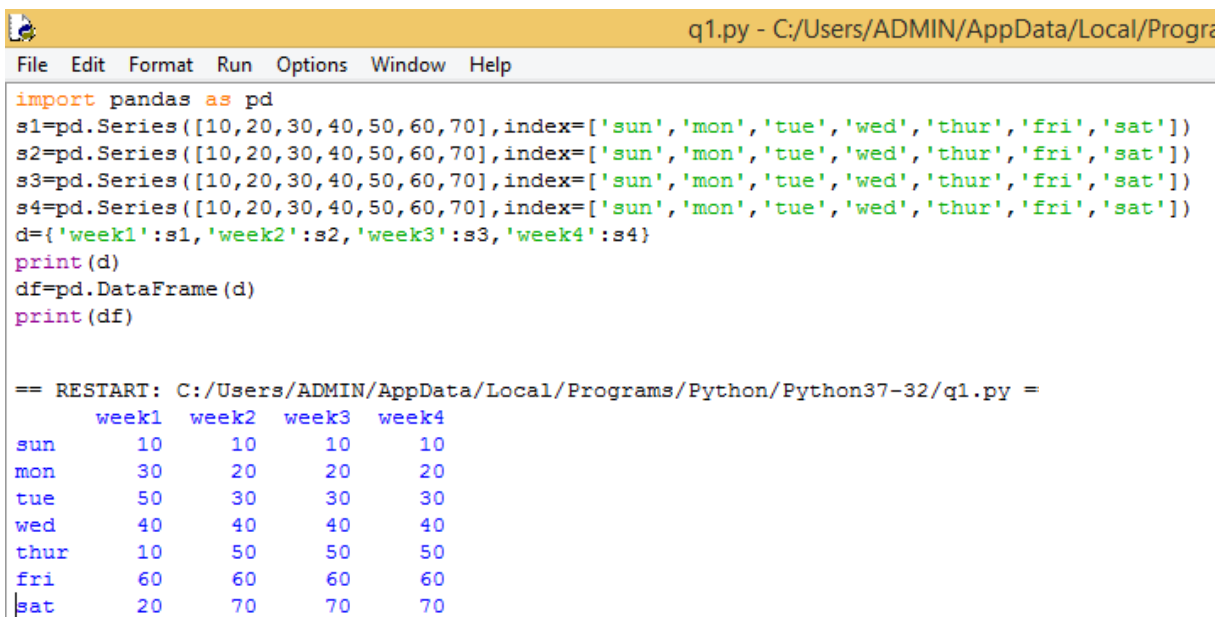
**Q.7 Why does the following code cause error :**

```
dfc1=pd.DataFrame([2,3,4])
dfc2=pd.DataFrame([[2,3,4]])
print(dfc1==dfc2)
```

**Ans:** Because both the dataframes are differently labeled.

It should be-

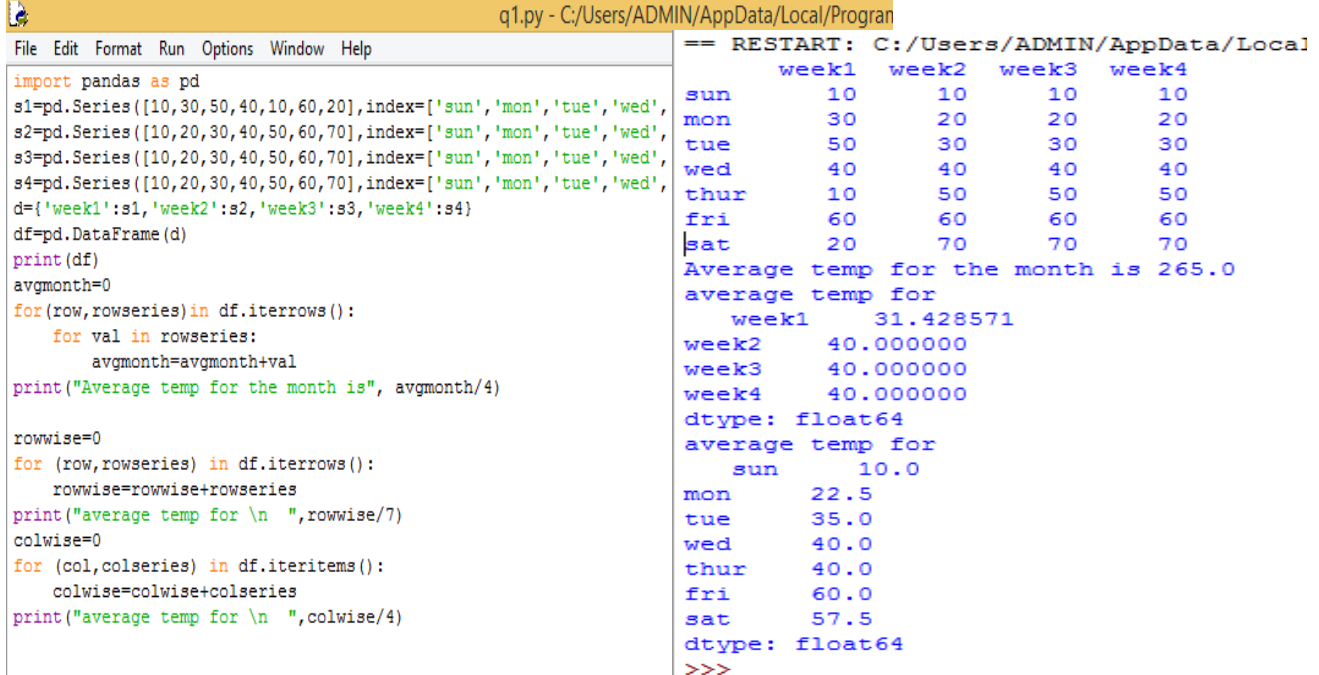
```
>>> df1=pd.DataFrame([2,3,4])
>>> df2=pd.DataFrame([2,3,4])
>>> print(df1==df2)
0
0 True
1 True
2 True
```

**Q.8 Four series objects Temp1, Temp2, Temp3 and Temp4 store the temperature of week1, week2, week3 and week4 respectively. Create a dataframe from these four series objects where the indexes should be 'Sunday', 'Monday', 'Tuesday' . . . 'Saturday', and columns should be 'Week1', 'Week2', 'Week3' and 'Week4'.****Ans:**


```
q1.py - C:/Users/ADMIN/AppData/Local/Programs/Python/Python37-32/
File Edit Format Run Options Window Help
import pandas as pd
s1=pd.Series([10,20,30,40,50,60,70],index=['sun','mon','tue','wed','thur','fri','sat'])
s2=pd.Series([10,20,30,40,50,60,70],index=['sun','mon','tue','wed','thur','fri','sat'])
s3=pd.Series([10,20,30,40,50,60,70],index=['sun','mon','tue','wed','thur','fri','sat'])
s4=pd.Series([10,20,30,40,50,60,70],index=['sun','mon','tue','wed','thur','fri','sat'])
d={'week1':s1,'week2':s2,'week3':s3,'week4':s4}
print(d)
df=pd.DataFrame(d)
print(df)

== RESTART: C:/Users/ADMIN/AppData/Local/Programs/Python/Python37-32/q1.py ==
   week1  week2  week3  week4
sun     10     10     10     10
mon     30     20     20     20
tue     50     30     30     30
wed     40     40     40     40
thur     10     50     50     50
fri     60     60     60     60
sat     20     70     70     70
```

- Q.9 From the dataframe object created in previous question, write a script to calculate –
- Average temperature for each day of the week i.e. average temperatures for Mondays, Tuesdays and so on.
  - Average temperature per week.
  - Average temperature of whole month.

Ans: 

```

q1.py - C:/Users/ADMIN/AppData/Local/Program
File Edit Format Run Options Window Help
import pandas as pd
s1=pd.Series([10,30,50,40,10,60,20],index=['sun','mon','tue','wed',
s2=pd.Series([10,20,30,40,50,60,70],index=['sun','mon','tue','wed',
s3=pd.Series([10,20,30,40,50,60,70],index=['sun','mon','tue','wed',
s4=pd.Series([10,20,30,40,50,60,70],index=['sun','mon','tue','wed',
d={'week1':s1,'week2':s2,'week3':s3,'week4':s4}
df=pd.DataFrame(d)
print(df)
avgmonth=0
for(row,rowseries)in df.iterrows():
    for val in rowseries:
        avgmonth=avgmonth+val
print("Average temp for the month is", avgmonth/4)

rowwise=0
for (row,rowseries) in df.iterrows():
    rowwise=rowwise+rowseries
print("average temp for \n ",rowwise/7)
colwise=0
for (col,colseries) in df.iteritems():
    colwise=colwise+colseries
print("average temp for \n ",colwise/4)

== RESTART: C:/Users/ADMIN/AppData/Local
week1 week2 week3 week4
sun 10 10 10 10
mon 30 20 20 20
tue 50 30 30 30
wed 40 40 40 40
thur 10 50 50 50
fri 60 60 60 60
sat 20 70 70 70
Average temp for the month is 265.0
average temp for
week1 31.428571
week2 40.000000
week3 40.000000
week4 40.000000
dtype: float64
average temp for
sun 10.0
mon 22.5
tue 35.0
wed 40.0
thur 40.0
fri 60.0
sat 57.5
dtype: float64
>>>

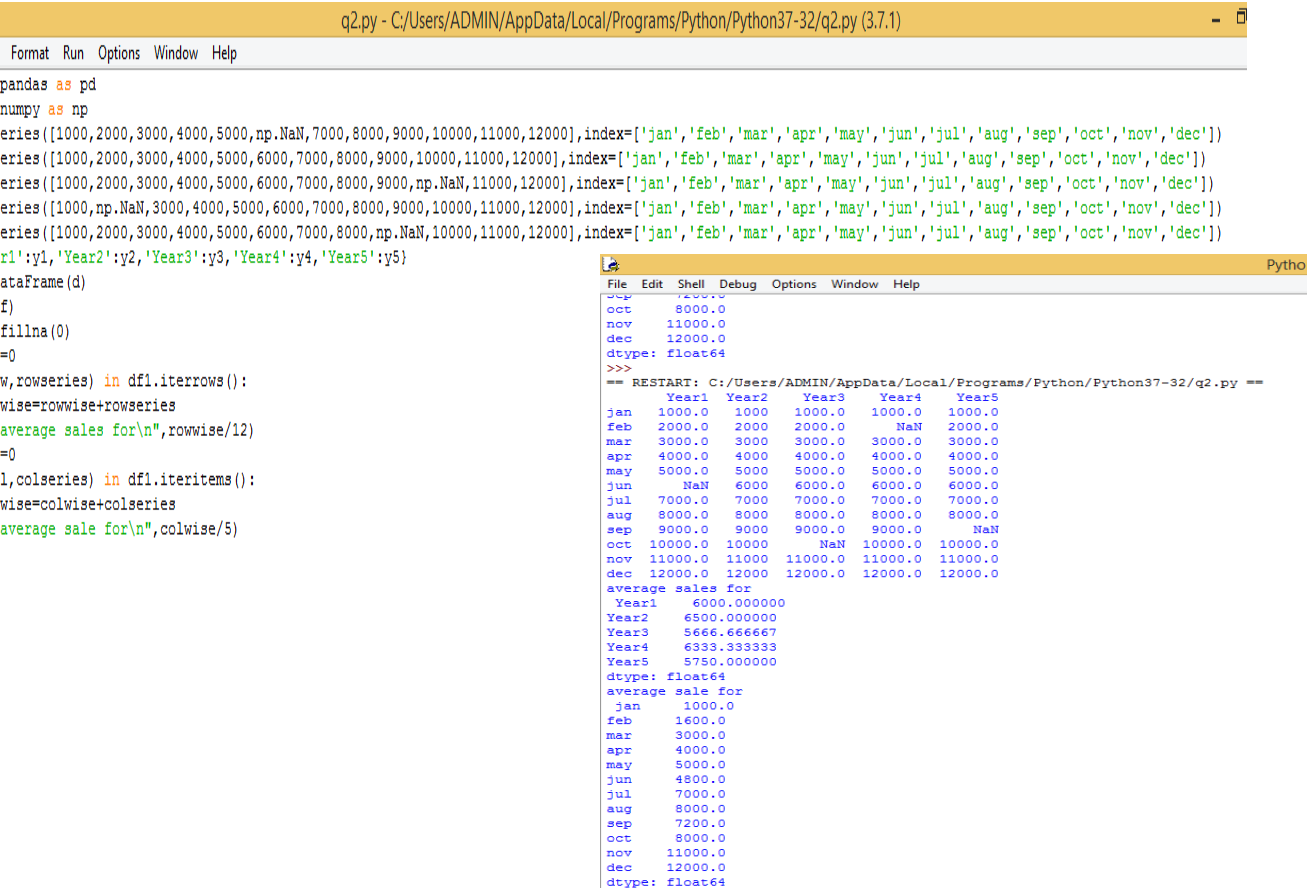
```

- Q.10 Given a dataframe that stores the details of past 5 years' monthly sales. Some old data is missing. Write a script to calculate average:

→ Monthly sales across the year → Yearly sales.

Make sure that missing values do not hamper the overall result.

Ans:



```

q2.py - C:/Users/ADMIN/AppData/Local/Programs/Python/Python37-32/q2.py (3.7.1)
File Edit Format Run Options Window Help
import pandas as pd
import numpy as np
y1=pd.Series([1000,2000,3000,4000,5000,np.NaN,7000,8000,9000,10000,11000,12000],index=['jan','feb','mar','apr','may','jun','jul','aug','sep','oct','nov','dec'])
y2=pd.Series([1000,2000,3000,4000,5000,6000,7000,8000,9000,10000,11000,12000],index=['jan','feb','mar','apr','may','jun','jul','aug','sep','oct','nov','dec'])
y3=pd.Series([1000,2000,3000,4000,5000,6000,7000,8000,9000,np.NaN,11000,12000],index=['jan','feb','mar','apr','may','jun','jul','aug','sep','oct','nov','dec'])
y4=pd.Series([1000,np.NaN,3000,4000,5000,6000,7000,8000,9000,10000,11000,12000],index=['jan','feb','mar','apr','may','jun','jul','aug','sep','oct','nov','dec'])
y5=pd.Series([1000,2000,3000,4000,5000,6000,7000,8000,np.NaN,10000,11000,12000],index=['jan','feb','mar','apr','may','jun','jul','aug','sep','oct','nov','dec'])
d={'Year1':y1,'Year2':y2,'Year3':y3,'Year4':y4,'Year5':y5}
df=pd.DataFrame(d)
print(df)
df1=df.fillna(0)
rowwise=0
for (row,rowseries) in df1.iterrows():
    rowwise=rowwise+rowseries
print("average sales for\n",rowwise/12)
colwise=0
for (col,colseries) in df1.iteritems():
    colwise=colwise+colseries
print("average sale for\n",colwise/5)

== RESTART: C:/Users/ADMIN/AppData/Local/Programs/Python/Python37-32/q2.py ==
Year1 Year2 Year3 Year4 Year5
jan 1000.0 1000 1000.0 1000.0 1000.0
feb 2000.0 2000 2000.0 NaN 2000.0
mar 3000.0 3000 3000.0 3000.0 3000.0
apr 4000.0 4000 4000.0 4000.0 4000.0
may 5000.0 5000 5000.0 5000.0 5000.0
jun NaN 6000 6000.0 6000.0 6000.0
jul 7000.0 7000 7000.0 7000.0 7000.0
aug 8000.0 8000 8000.0 8000.0 8000.0
sep 9000.0 9000 9000.0 9000.0 NaN
oct 10000.0 10000 NaN 10000.0 10000.0
nov 11000.0 11000 11000.0 11000.0 11000.0
dec 12000.0 12000 12000.0 12000.0 12000.0
average sales for
Year1 6000.000000
Year2 6500.000000
Year3 5666.666667
Year4 6333.333333
Year5 5750.000000
dtype: float64
average sale for
jan 1000.0
feb 1600.0
mar 3000.0
apr 4000.0
may 5000.0
jun 4800.0
jul 7000.0
aug 8000.0
sep 7200.0
oct 8000.0
nov 11000.0
dec 12000.0
dtype: float64
>>>

```

## Chapter – 12: DATA TRANSFER BETWEEN FILES, SQL, DATABASES AND DATAFRAMES

### Very Short answer Type Questions

**Q.1 What all libraries do you require in order to bring data from CSV file into a dataframe?**

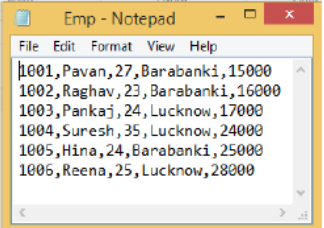
**Ans:** Pandas and Sqlite3 Library

**Q.2 You want to read data from a CSV file in a dataframe but you want to provide your own column names to dataframe. What additional argument would you specify in read\_csv( )?**

**Ans:** names, for ex-

```
>>> empdata=pd.read_csv("C:\\data\\emp.csv", names=['RollNo.', 'EmpName', 'Age', 'City', 'Salary'])
>>> empdata
```

RollNo.	EmpName	Age	City	Salary	
0	1001	Pavan	27	Barabanki	15000
1	1002	Raghav	23	Barabanki	16000
2	1003	Pankaj	24	Lucknow	17000
3	1004	Suresh	35	Lucknow	24000
4	1005	Hina	24	Barabanki	25000
5	1006	Reena	25	Lucknow	28000



**Q.3 Which argument would you give to read\_csv( ) if you only want to read top 10 rows of data?**

**Ans:** nrows, for ex-

```
>>> empdata=pd.read_csv("C:\\data\\emp.csv",\
                        names=['RollNo.', 'EmpName', 'Age', 'City', 'Salary'],nrows=3)
>>> empdata
```

RollNo.	EmpName	Age	City	Salary	
0	1001	Pavan	27	Barabanki	15000
1	1002	Raghav	23	Barabanki	16000
2	1003	Pankaj	24	Lucknow	17000

Use of nrows= <n>

**Q.4 Write command to store data of dataframe mdf into a CSV file Mydata.csv, with separate character as '@'.**

**Ans:** mdf.to\_csv("D:\\Mydata.csv",sep='@')

### Short Answer Type Questions

**Q.1 What are advantages of CSV file formats?**

**Ans:** advantages:

1. A Simple, compact and ubiquitous format for data storage.
2. A common format for data interchange.
3. It can be opened in popular spreadsheet packages like MS-Excel, Cal etc.
4. Nearly all spradsheets and databases support import/export to csv format.

**Q.2 By default, read\_csv( ) uses the values of the first row as column headers in dataframes. Which argument will you give to ensure that the top/first row's data is used as data and not as column headers?**

**Ans:** header=None, for ex- mdf=pd. read\_csv("D:\\Mydata.csv",header=None)

**Q.3 Why do you need connection to an SQL database in order to get data from a table?**

**Ans:** A connection works like a channel connecting Python and the database file through which data can travel to and from SQL database.

**Q.4 What is sqlite3 library of Python?**

**Ans:** The sqlite3 library deals with communication with SQLite databases. Sqlite3 is by default bundled with Python.

**Q.5 What all libraries do you require in order to interact with sqlite3 databases (and dataframe) from within Python?**

**Ans:** Pandas and Sqlite3 Library

**Q.6 What additional argument do you need to specify in to\_sql( ) so that old data of SQL table is retained?**

**Ans:** append for ex- df.to\_sql("metros",conn, if\_exists="append")

**Q.7 If query is a string storing an SQL statement. Write statements so that the data is fetched based on query from SQL database Mydata.db.**

**Ans:** df=pd.read\_sql("select \* from stud",conn)

### Skill Based Questions

**Q.1 WAP to read details such as item, sales made in a dataframe and then store this data in a CSV file.**

**Ans:**

```
File Edit Format Run Options Window Help
import pandas as pd
d={'Fridge':[12], 'Cooker':[5], 'Juicer':[15], 'Iron':[11]}
df=pd.DataFrame(d)
print(df)
df.to_csv("D:\\file.csv")
```

**Q.2 WAP to read data from a CSV file where separator character is '@'. Make sure that:  
→ The top row is used as data, not as column headers.**

```
import pandas as pd
df=pd.read_csv("D:\\read.csv",sep='@', header=None)
print(df)
```

```
Python 3.7.1
File Edit Shell Debug Options Window Help
Python 3.7.1 (v3.7.1:260ec2c36a, Oct 20 201
l) on win32
Type "help", "copyright", "credits" or "lic
>>>
RESTART: C:\Users\ADMIN\AppData\Local\Progr
file.py
      0      1      2
0  Neha  31000  32
1  Surjeet  12000  34
2  Anand  21000  43
3  Juhi  14000  21
```

→ Only 20 rows are read into dataframe.

```
reading from file.py - C:\Users\ADMIN\AppData\Local\Programs\Python\Python37-32\reading from file.py (3.7.1)
File Edit Format Run Options Window Help
import pandas as pd
df=pd.read_csv("D:\\read.csv",sep='@', header=None,nrows=20)
print(df)
```

Q.3 WAP to input the details of a students through Python and put those values into student table of Mydata.db.

Ans:

```
File Edit Format Run Options Window Help
import pandas as pd
import sqlite3 as sq
dict={'Name':['Neha','Shikha','Priya'],'Marks':[76.8,74.9,87.55]}
df=pd.DataFrame(dict)
print(df)
conn=sq.connect("C:\\sqlite\\Mydata.db")
df.to_sql("class",conn)
```

## Chapter – 13: RELATIONAL DATABASES

### Short Answer Type Questions

**Q.1 What is a database system? What is its need?**

**Ans:** A database is a collection of information that is organized so that it can be easily accessed, managed and updated. Data is organized into rows, columns and tables, and it is indexed to make it easier to find relevant information. Data gets updated, expanded and deleted as new information is added. Databases process workloads to create and update themselves, querying the data they contain and running applications against it. The software which facilitates the collection of database and its retrieval is known as DBMS. A database along with a DBMS is referred to as a database system.

DBMS is needed to overcome the following problems in traditional file system –

- (i) Data redundancy
- (ii) Data inconsistency
- (iii) Unstandardized data
- (iv) Insecure data
- (v) Incorrect data etc.

In DBMS these limitations has been almost reduced.

**Q.2 What are the disadvantages of database systems?**

**Ans:** When we work with complex task then there may be some disadvantages of using databases these are –

- (i) Security may be compromised without good controls.
- (ii) Integrity may be compromised without good controls.
- (iii) Extra hardware may be required.
- (iv) Performance overhead may be significant.
- (v) System is likely to be complex.

**Q.3 What are views? How they are useful?**

**Ans:** A view is a virtual table that does not really exist in its own right but is instead derived from one or more underlying base tables(s). the view is a kind of table whose contents are taken upon other tables depending upon a given query condition. No stored file is created to store the contents of a view rather its definition is stored only.

---

The usefulness of views lies in the fact that they provide an excellent way to give people access to some but not all of the information in a table.

#### **Q.4 What is the purpose of using MySQL?**

**Ans:** MySQL is a free, open source *Relational Database Management System* (RDBMS) that uses Structured Query Language (SQL). In MySQL database, information is stored in *Tables*. A single MySQL database can contain many tables at once and store thousands of individual records.

#### **Q.5 What do you understand by MySQL server instance?**

**Ans:** MySQL server instance is created from background processes and applications. It stays in memory and listens for client requests coming in over the network and accesses database contents according to those requests and provides that to the clients.

#### **Q.6 What do you understand by MySQL client?**

**Ans:** MySQL clients are programs that connect to the MySQL server and issue queries in a pre-specified format. MySQL is compatible with the standards based SQL. The client program may contact the server programmatically or manually.

#### **Q.7 What is SQL? What are different categories of commands available in SQL?**

**Ans:** In order to access the data within Oracle databases, all programs and users must use, Structured Query Language (SQL). SQL is the set of commands that is recognized by nearly all RDBMSs. SQL commands can be divided into following categories –

- (i) Data Definition Language (DDL) Commands.
- (ii) Data Manipulation Language (DML) Commands.
- (iii) Transaction Control Language (TCL) Commands.
- (iv) Session Control Commands.
- (v) System Control Commands.

#### **Q.8 Differentiate between DDL and DML commands.**

**Ans:** **The DDL commands**, as the name suggests, allow you to perform tasks related to data definition i.e. through these commands you can perform tasks like create, alter and drop schema objects, grant and revoke privileges etc.

**The DML commands**, as the name suggests, are used to manipulate data i.e. DML commands query and manipulate data in existing schema objects like, Select query, insert into command etc.

#### **Q.9 What is MySQL? What are its functions and features?**

**Ans:** MySQL is a free, open source *Relational Database Management System* (RDBMS) that uses Structured Query Language (SQL). In MySQL database, information is stored in *Tables*. A single MySQL database can contain many tables at once and store thousands of individual records.

Its functions and features are –

- (i) Speed
- (ii) Ease of use
- (iii) Cost
- (iv) Query Language support
- (v) Portability
- (vi) Data Types
- (vii) Security
- (viii) Scalability and Limits



- (ix) Connectivity
- (x) Localization
- (xi) Clients and Tools

**Q.10 Name some commands used to assign/revoke privileges from database users.**

**Ans:** Grant Privilege Command and Revoke Privilege commands.

**Q.11 Name some Table maintenance commands?**

**Ans:** ANALYZE TABLE , CHECK TABLE , CHECKSUM TABLE , OPTIMIZE TABLE , REPAIR TABLE

## Chapter – 17: SIMPLE QUERIES IN SQL

### Very Short Answer Type Questions

**Q.1 Name some basic MySQL SQL elements.**

**Ans:** Some basic MySQL SQL elements are : (i) Literals (ii) Datatypes (iii) Nulls (iv) Comments.

**Q.2 Differentiate between CHAR and VARCHAR Datatypes.**

**Ans:** The difference between CHAR and VARCHAR is that of *fixed length* and *variable length* respectively.

**Q.3 Write two usage of DESC in SQL.**

**Ans:** DESC is used with Order by Clause in select statement which displays the data in descending order. And another use of DESC is that it is used to display the structure of table.

**Q.4 Consider the following Table named 'empl' and Write SQL commands from (a) to (n)**

empno	ename	job	mgr	hiredate	sal	comm	deptno
8369	SMITH	CLERK	8902	1990-12-18	800.00	NULL	20
8499	ANYA	SALESMAN	8698	1991-02-20	1600.00	300.00	30
8521	SETH	SALESMAN	8698	1991-02-22	1250.00	500.00	30
8566	MAHADEVAN	MANAGER	8839	1991-04-02	2985.00	NULL	20
8654	MOMIN	SALESMAN	8698	1991-09-28	1250.00	1400.00	30
8698	BINA	MANAGER	8839	1991-05-01	2850.00	NULL	30
8882	SHIAVNSH	MANAGER	8839	1991-06-09	2450.00	NULL	10
8888	SCOTT	ANALYST	8566	1992-12-09	3000.00	NULL	20
8839	AMIR	PRESIDENT	NULL	1991-11-18	5000.00	NULL	10
8844	KULDEEP	SALESMAN	8698	1991-09-08	1500.00	0.00	30
8886	ANOOP	CLERK	8888	1993-01-12	1100.00	NULL	20
8900	JATIN	CLERK	8698	1991-12-03	950.00	NULL	30
8902	FAKIR	ANALYST	8566	1991-12-03	3000.00	NULL	20
8934	MITA	CLERK	8882	1992-01-23	1300.00	NULL	10

**(a) Display all the records from table empl.**

**Ans:** select \* from empl;

**(b) Display EmpNo and EName of all employees from the table empl.**

**Ans:** select EmpNo, EName from empl;

**(c) Display employee name, salary, and department number who are not getting commission from table empl.**

**Ans:** select EName, sal, deptno from empl where comm = NULL;

**(d) Display employee number, name, sal\*12 as Annual salary whose commission is not NULL from table empl.**

**Ans:** select EmpNo, EName, sal\*12 'Annual Salary' from empl where sal<>NULL;

**(e) List all department numbers from table empl.**

**Ans:** select deptno from empl;

**(f) List all unique department numbers from table empl.**

**Ans:** select distinct(deptno) from empl;

**(g) List the details of all clerks who have not been assigned department as yet.**

**Ans:** select \* from empl where job='CLERK' and deptno=NULL;

**(h) Display the details of employees whose name have only four letters.**

**Ans:** select \* from empl where EName like '\_\_\_\_';

**(i) Display the details of all employee whose annual salary is between 25000 to 40000.**

**Ans:** select \* from empl where sal between 25000 and 40000;

**(j) How many job types are offered to employees?**

**Ans:** select count(distinct(job)) from empl;

**(k) List the employees who earn more commission than their salaries.**

**Ans:** select \* from empl where sal<comm;

**(l) Display name, job title and salary of employee who do not have manager.**

**Ans:** select EName, job, sal from empl where mgr = NULL;

**(m) Display the name of employee whose name contains 'A' as third letter.**

**Ans:** select EName from empl where EName like '\_\_\_A%'

**(n) Display the name of employee whose name contains 'L' as any letter.**

**Ans:** select EName from empl where EName like '%L%'

Consider the above table 'empl' and give answer-

**Q.5 Write a query to display EName and Sal of employee whose salary is greater than or equal to 2200 from table empl.**

**Ans:** select EName, sal from empl where sal>=2200;

 Write SQL commands for the following on the basis of given table STUDENT :

Table : STUDENT1

No.	Name	Stipend	Stream	AvgMark	Grade	Class
1	Karan	400.00	Medical	78.5	B	12B
2	Divakar	450.00	Commerce	89.2	A	11C
3	Divya	300.00	Commerce	68.6	C	12C
4	Arun	350.00	Humanities	73.1	B	12C
5	Sabina	500.00	Nonmedical	90.6	A	11A
6	John	400.00	Medical	75.4	B	12B
7	Robert	250.00	Humanities	64.4	C	11A
8	Rubina	450.00	Nonmedical	88.5	A	12A
9	Vikas	500.00	Nonmedical	92.0	A	12A
10	Mohan	300.00	Commerce	67.5	C	12C

**Ans:** (a) Select \* from CLUB;

(a) Select COACHNAME, DATEOFAPP from CLUB order by DATEOFAPP desc;

(b) Select COACHNAME, PAY, AGE, PAY\*15/100 'BONUS' from CLUB;

Q.6 Write SQL commands for the following on the basis of given table CLUB :

Table : CLUB

COACH_ID	COACHNAME	AGE	SPORTS	DATOFAPP	PAY	SEX
1.	KUKREJA	35	KARATE	27/03/1996	1000	M
2.	RAVINA	34	KARATE	20/01/1998	1200	F
3.	KARAN	34	SQUASH	19/02/1998	2000	M
4.	TARUN	33	BASKETBALL	01/01/1998	1500	M
5.	ZUBIN	36	SWIMMING	12/01/1998	750	M
6.	KETAKI	36	SWIMMING	24/02/1998	800	F
7.	ANKITA	39	SQUASH	20/02/1998	2200	F
8.	ZAREEN	37	KARATE	22/02/1998	1100	F
9.	KUSH	41	SWIMMING	13/01/1998	900	M
10.	SHAILYA	37	BASKETBALL	19/02/1998	1700	M

(a) To show all information about the swimming coaches in the club.

(b) To list names of all coaches with their date of appointment (DATOFAPP) in descending order.

(c) To display a report, showing coachname, pay, age and bonus (15% of pay) for all the coaches.

**Ans:** (a) select \* from STUDENT1 where Stream = 'NONMEDICAL';

(b) select Name from STUDENT1 where CLASS like '12\_' order by Stipend;

(c) select \* from STUDENT1 order by AvgMark desc;

(d) select Name, Stipend, Stream, Stipend\*12 'Stipend in a Year' from STUDENT1;

## Chapter – 18: TABLE CREATION AND DATA MANIPULATION COMMANDS

### Short Answer Type Questions

**Q.1** Using SQL statements in MySQL, create the tables identified bellow in the following order.

**Database Name: Univ.**

**Create database if needed. (Underlined Columns depict primary key)**

**Campus** (CampusID, CampusName, Street, City, State, Pin, Phone, CampusDiscount)

**Position** (PositionID, Position, YearlyMembershipFee)

**Members** (MemberID, Lastname, FirstName, CampusAddress, CampusPhone, CampusID, PositionID, ContractDuration)

**Foreign Keys** CampusID → Campus(CampusID)

PositionID → Position(PositionID)

**Ans:** mysql>CREATE DATABASE Univ;

mysql>USE Univ;

mysql>CREATE TABLE Campus(CampusID VARCHAR(5) PRIMARY KEY,

```

CampusName VARCHAR(20),
Street VARCHAR(20),
City VARCHAR(20),
State VARCHAR(15),
Pin INT(6),
Phone VARCHAR(13),
CampusDiscount INT(2));

mysql>CREATE TABLE Position (PositionID VARCHAR(4) PRIMARY KEY,
Position VARCHAR(10),
YearlyMemberShipFee INT(4));

mysql>CREATE TABLE Members (MemberID VARCHAR(5) PRIMARY KEY,
LastName VARCHAR(15),
FirstName VARCHAR(15),
CampuAddress VARCHAR(30),
CampusPhone VARCHAR(13),
CAMPUSID VARCHAR(5) REFERENCES Campus(CampusID),
PositionID VARCHAR(4) REFERENCES Position(PositionID),
ContractDuration INT(2));

```

## Q.2 Write SQL commands to perform the following tasks –

(a) Create table Employee with the following structure:

Name of Column	ID	First_Name	Last_Name	User_ID	Salary
Type	Number(4)	Varchar(30)	Varchar(30)	Varchar(10)	Number(9,2)

Ensure the following specification in created table:

- ✚ ID should be declared as Primary Key
- ✚ User\_ID should be unique
- ✚ Salary Must be greater than 5000
- ✚ First\_Name and Lst\_Name must not remain Blank

**Ans:** mysql>CREATE TABLE Employee (ID NUMBER(4) PRIMARY KEY,  
First\_Name VARCHAR(30) NOT NULL,  
Last\_Name VARCHAR(30) NOT NULL,  
User\_ID VARCHAR(10) UNIQUE,  
Salary NUMBER(9,2));

(b) Create another table Job with following specification:

Name of Column	Type
Job_ID	Number(4)
Job_des	Varchar(30)
Alloc_on	Date
Due_on	Date
Emp_id	Number(4)

Ensure the following specification in created table:

- ✚ Job\_ID should be declared as Primary Key
- ✚ Job\_des, Alloc\_on, Due\_on cannot be left blank
- ✚ Emp\_id is foreign key here that us related to ID column of earlier created table Employee.

---

**Ans:** mysql>CREATE TABLE Job (Job\_ID NUMBER(4) PRIMARY KEY,  
Job\_des VARCHAR(30) NOT NULL,  
Alloc\_on DATE NOT NULL,  
Due\_on DATE NOT NULL,  
Emp\_id NUMBER(4) REFERENCES Employee(ID));

**(a) Show the structure of the table employee.**

**Ans:** mysql>DESC Employee;

**(b) Show the structure of the table job.**

**Ans:** mysql>DESC Job;

**(c) Insert 10 records into the table employee as well as Job.**

**Ans:** mysql> INSERT INTO Employee VALUES(1, 'Amit', 'Kumar', 'E1001',20000);

**Same remaining 9 values can be inserted into Employee Table.**

mysql> INSERT INTO Job VALUES(1001, 'Manager', '12-25-2016', '12-28-2017', 1);

**Same remaining 9 values can be inserted into Job Table.**

**(d) For record with ID 4 update record with Last\_Name, User\_ID and Salary.**

**Ans:** mysql>UPDATE Employee SET Last\_Name='Singh' WHERE ID=4;

mysql>UPDATE Employee SET Salary=25000 WHERE ID=4;

mysql>UPDATE Employee SET User\_ID='E1004' WHERE ID=4;

**(e) Make the changes permanent.**

**Ans:** mysql>COMMIT

**(f) Modify the salary and increase it by 1000, for all who get salary less than 6000.**

**Ans:** mysql>UPDATE Employee SET Salary = Salary+1000 WHERE Salary<6000;

**(g)Add one new column in table Employee named 'Phone'.**

**Ans:** mysql>ALTER TABLE Employee ADD(Phone VARCHAR(13));

**(h)Provide the phone numbers to all the employees.**

**Ans:** mysql>INSERT INTO Employee (Phone) VALUES('+918888888888');

**Same remaining 9 values can be inserted into Employee Table.**

**(i) Delete the Employee record having First\_Name as Manish.**

**Ans:** mysql>DELETE FROM Employee WHERE First\_Name='Manish';

**(j) Remove employee table permanently.**

**Ans:** mysql>DROP TABLE Employee;

**(k)Count the total records in the table employee.**

**Ans:** mysql>SELECT Count(\*) FROM Employee;

**Q.3 What are the different divisions of SQL and commands? Give examples.**

**Ans:** SQL commands can be divided into following categories –

- (i) Data Definition Language (DDL) Commands. e.g. CREATE TABLE, ALTER TABLE, DROP TABLE etc.
- (ii) Data Manipulation Language (DML) Commands. e.g. INSERT INTO, DELETE FROM, UPDATE, SELECT etc.
- (iii) Transaction Control Language (TCL) Commands.e.g. ROLLBACK, COMMIT, SAVEPOINT etc.

**Q.4 What is foreign key? How do you define a foreign key in your table?**

**Ans:** A foreign key is a column or group of columns in a relational database table that provides a link between data in two tables. It acts as a cross-reference between tables because it references the primary key of another table, thereby establishing a link between them.

To define a foreign key in a table we have to use REFERENCES keyword as follows -

```
mysql>CREATE TABLE Job (Job_ID NUMBER(4) PRIMARY KEY,
                        Job_des VARCHAR(30) NOT NULL,
                        Alloc_on DATE NOT NULL,
                        Due_on DATE NOT NULL,
                        Emp_id NUMBER(4) REFERENCES Employee(ID));
```

In the above example Emp\_id is a foreign key which references the ID field of table Employee.

**Q.5 How foreign key command is different from Primary Key command?**

**Ans:** A primary key is a special key in a relational database that acts as a unique identifier for each record meaning it uniquely identifies each row/record in a table and its value should be unique for each row of the table. A foreign key, on the other hand, is a field in one table that link two tables together. It refers to a column or a group of columns that uniquely identifies a row of another table or same table.

A primary key is a combination of UNIQUE and Not Null constraints so no duplicate values can be allowed to have in a primary key field in a relational database table. No two rows are allowed to carry duplicate values for a primary key attribute. Unlike a primary key, foreign key can contain duplicate values and a table in a relational database can contain more than foreign key.

**Q.6 How is Foreign Key commands related to Primary Key?**

**Ans:** A foreign key refers to a field or a collection of fields in a database record that uniquely identifies a key field of another database record in some other table. In simple terms, it establishes a link between records in two different tables in a database. It can be a column in a table that points to the primary key columns meaning a foreign key defined in a table refers to the primary key of some other table. References are crucial in relational databases to establish links between records which is essential for sorting databases. Foreign keys play an important role in relational database normalization especially when tables need to access other tables.

**Q.7 How do you enforce business rules on a database?**

**Ans:** We can enforce Business Rules in Relational Databases Using Constraints. Constraints are restrictions over a column. Or we can say that constraint is a check or condition applicable on a field or set of fields. These are of following types -

- (i) Unique
- (ii) Primary Key
- (iii) Default
- (iv) Check
- (v) Foreign Key

**Q.8 What are table constraints? What are column constraints? How these two are different?**

**Ans:** Table constraints apply to groups of one or more columns whereas column constraints apply only to Individual column.



---

**Q.9 What is default value? How do you define it? What is the default value of column for which no default value is defined?**

**Ans:** A default value can be specified for a column using DEFAULT clause. When a user does not enter a value for the column (having default value), automatically the defined default value is inserted in the field. e.g.

```
mysql>CREATE TABLE Employee (ID NUMBER(4) PRIMARY KEY,  
                                First_Name VARCHAR(30) NOT NULL,  
                                Last_Name VARCHAR(30) NOT NULL,  
                                User_ID VARCHAR(10) UNIQUE,  
                                Salary NUMBER(9,2) DEFAULT 15000);
```

**Q.10 Differentiate between -**

**(i) DROP TABLE, DROP DATABASE**

**(ii) DROP TABLE, DROP clause of ALTER TABLE.**

**Ans:** (i) DROP TABLE will delete a table whereas DROP DATABASE will delete a complete database which will contain different tables. These are DDL commands.

**Q.11 How will you enlist the name of tables in a database?**

**Ans:** By using SHOW TABLES command. e.g.

```
mysql>SHOW TABLES;
```

**Q.12 How to view the structures of the table made by you?**

**Ans:** By using DESC <TableName> command. e.g.

```
mysql>DESC Employee;
```

**Q.13 How to Drop Table and Databases?**

**Ans:** By using DROP TABLE command. e.g.

```
mysql>DROP TABLE Employee;
```

**Please Visit: [www.pythontrends.wordpress.com](http://www.pythontrends.wordpress.com)**

**YouTube Channel: Python Trends**

**Developed By:**

**Sanjeev Bhadauria PGT CS, KV Barabanki**

**Neha Tyagi PGT CS, KV no-5 Jaipur**