**Python Exercise**

**1.**Write a Python program to print the following string in a specific format (see the output).  
*Sample String :* "Twinkle, twinkle, little star, How I wonder what you are! Up above the world so high, Like a diamond in the sky. Twinkle, twinkle, little star, How I wonder what you are"  
*Output :*

Twinkle, twinkle, little star,

How I wonder what you are!

Up above the world so high,

Like a diamond in the sky.

Twinkle, twinkle, little star,

How I wonder what you are

**2.**Write a Python program to find out what version of Python you are using.

**3.**Write a Python program to display the current date and time.  
*Sample Output :*  
Current date and time :  
2014-07-05 14:34:14

**4.**Write a Python program that calculates the area of a circle based on the radius entered by the user.  
*Sample Output :*  
r = 1.1  
Area = 3.8013271108436504

**5.** Write a Python program that accepts the user's first and last name and prints them in reverse order with a space between them.

**6.** Write a Python program that accepts a sequence of comma-separated numbers from the user and generates a list and a tuple of those numbers.  
*Sample data :*3, 5, 7, 23  
*Output :*  
List : ['3', ' 5', ' 7', ' 23']  
Tuple : ('3', ' 5', ' 7', ' 23')

**7.** Write a Python program that accepts a filename from the user and prints the extension of the file.  
*Sample filename :* abc.java  
*Output :* java

**8.** Write a Python program to display the first and last colors from the following list.  
color\_list = ["Red","Green","White" ,"Black"]

**9.** Write a Python program to display the examination schedule. (extract the date from exam\_st\_date).  
exam\_st\_date = (11, 12, 2014)  
Sample Output : The examination will start from : 11 / 12 / 2014

**10.**Write a Python program that accepts an integer (n) and computes the value of n+nn+nnn.  
*Sample value of n is*5 *Expected Result :*615

**11.** Write a Python program to print the documents (syntax, description etc.) of Python built-in function(s).  
*Sample function*: abs() *Expected Result*:  
abs(number) -> number  
Return the absolute value of the argument.

**12.** Write a Python program that prints the calendar for a given month and year.  
*Note :*Use 'calendar' module.

**13.** Write a Python program to print the following 'here document'.  
*Sample string*:  
a string that you "don't" have to escape  
This  
is a ....... multi-line  
heredoc string --------> example

**14.** Write a Python program to calculate the number of days between two dates.  
*Sample dates* : (2014, 7, 2), (2014, 7, 11)  
*Expected output*: 9 days

**15.** Write a Python program to get the volume of a sphere with radius six.

**16.** Write a Python program to calculate the difference between a given number and 17. If the number is greater than 17, return twice the absolute difference.

**17.** Write a Python program to test whether a number is within 100 of 1000 or 2000.

**18.** Write a Python program to calculate the sum of three given numbers. If the values are equal, return three times their sum

**19.** Write a Python program to get a newly-generated string from a given string where "Is" has been added to the front. Return the string unchanged if the given string already begins with "Is".

**20.** Write a Python program that returns a string that is n (non-negative integer) copies of a given string.

**21.** Write a Python program that determines whether a given number (accepted from the user) is even or odd, and prints an appropriate message to the user.

**22.** Write a Python program to count the number 4 in a given list.

**23.** Write a Python program to get n (non-negative integer) copies of the first 2 characters of a given string. Return n copies of the whole string if the length is less than 2.

**24.** Write a Python program to test whether a passed letter is a vowel or not.

**25.** Write a Python program that checks whether a specified value is contained within a group of values.  
*Test Data* :  
3 -> [1, 5, 8, 3] : True  
-1 -> [1, 5, 8, 3] : False

**26.** Write a Python program to create a histogram from a given list of integers.

**27.** Write a Python program that concatenates all elements in a list into a string and returns it.

**28.** Write a Python program to print all even numbers from a given list of numbers in the same order and stop printing any after 237 in the sequence.  
*Sample numbers list* :

numbers = [

386, 462, 47, 418, 907, 344, 236, 375, 823, 566, 597, 978, 328, 615, 953, 345,

399, 162, 758, 219, 918, 237, 412, 566, 826, 248, 866, 950, 626, 949, 687, 217,

815, 67, 104, 58, 512, 24, 892, 894, 767, 553, 81, 379, 843, 831, 445, 742, 717,

958,743, 527

]

**29.** Write a Python program that prints out all colors from color\_list\_1 that are not present in color\_list\_2.  
*Test Data*:  
color\_list\_1 = set(["White", "Black", "Red"])  
color\_list\_2 = set(["Red", "Green"])  
*Expected Output*:  
{'Black', 'White'}

**30.** Write a Python program that will accept the base and height of a triangle and compute its area.

**31.** Write a Python program that computes the greatest common divisor (GCD) of two positive integers.

**32.** Write a Python program to find the least common multiple (LCM) of two positive integers.

**33.** Write a Python program to sum three given integers. However, if two values are equal, the sum will be zero.

**34.** Write a Python program to sum two given integers. However, if the sum is between 15 and 20 it will return 20.

**35.** Write a Python program that returns true if the two given integer values are equal or their sum or difference is 5.

**36.** Write a Python program to add two objects if both objects are integers.

**37.** Write a Python program that displays your name, age, and address on three different lines.

**38.** Write a Python program to solve (x + y) \* (x + y).  
*Test Data* : x = 4, y = 3  
*Expected Output* : (4 + 3) ^ 2) = 49

**39.** Write a Python program to compute the future value of a specified principal amount, rate of interest, and number of years.  
*Test Data* : amt = 10000, int = 3.5, years = 7  
*Expected Output* : 12722.79

**40.** Write a Python program to calculate the distance between the points (x1, y1) and (x2, y2).

**41.** Write a Python program to check whether a file exists.

**42.** Write a Python program to determine if a Python shell is executing in 32bit or 64bit mode on OS.

**43.** Write a Python program to get OS name, platform and release information.

**44.** Write a Python program to locate Python site packages.

**45.**Write a Python program that calls an external command.

**46.** Write a Python program to retrieve the path and name of the file currently being executed.

**47.** Write a Python program to find out the number of CPUs used.

**48.** Write a Python program to parse a string to float or integer.

**49.** Write a Python program to list all files in a directory.

**50.** Write a Python program to print without a newline or space.

**51.** Write a Python program to determine the profiling of Python programs.  
Note: A profile is a set of statistics that describes how often and for how long various parts of the program executed. These statistics can be formatted into reports via the pstats module.

**52.** Write a Python program to print to STDERR.

**53.** Write a Python program to access environment variables.

**54.** Write a Python program to get the current username.

**55.** Write a Python program to find local IP addresses using Python's stdlib.

**56.** Write a Python program to get the height and width of the console window.

**57.**Write a Python program to get the execution time of a Python method.

**58.**Write a Python program to sum the first n positive integers.

**59.**Write a Python program to convert height (in feet and inches) to centimeters.

**60.**Write a Python program to calculate the hypotenuse of a right angled triangle.

**61.**Write a Python program to convert the distance (in feet) to inches, yards, and miles.

**62.**Write a Python program to convert all units of time into seconds.

**63.**Write a Python program to get an absolute file path.

**64.**Write a Python program that retrieves the date and time of file creation and modification.

**65.**Write a Python program that converts seconds into days, hours, minutes, and seconds

**66.**Write a Python program to calculate the body mass index.

**67.**Write a Python program to convert pressure in kilopascals to pounds per square inch, a millimeter of mercury (mmHg) and atmosphere pressure.

**68.**Write a Python program to calculate sum of digits of a number.

**69.**Write a Python program to sort three integers without using conditional statements and loops.

**70.**Write a Python program to sort files by date.

**71.**Write a Python program to get a directory listing, sorted by creation date.

**72.**Write a Python program to get the details of the math module.

**73.**Write a Python program to calculate the midpoints of a line.

**74.**Write a Python program to hash a word.

**75.**Write a Python program to get the copyright information and write Copyright information in Python code.

**76.**Write a Python program to get the command-line arguments (name of the script, the number of arguments, arguments) passed to a script.

**77.**Write a Python program to test whether the system is a big-endian platform or a little-endian platform.

**78.**Write a Python program to find the available built-in modules.

**79.**Write a Python program to get the size of an object in bytes.

**80.**Write a Python program to get the current value of the recursion limit.

**81.**Write a Python program to concatenate N strings.

**82.**Write a Python program to calculate the sum of all items of a container (tuple, list, set, dictionary).

**83.**Write a Python program to test whether all numbers in a list are greater than a certain number.

**84.**Write a Python program to count the number of occurrences of a specific character in a string.

**85.**Write a Python program to check whether a file path is a file or a directory.

**86.**Write a Python program to get the ASCII value of a character.

**87.**Write a Python program to get the size of a file.

**88.**Given variables x=30 and y=20, write a Python program to print "30+20=50".

**89.**Write a Python program to perform an action if a condition is true.  
Given a variable name, if the value is 1, display the string "First day of a Month!" and do nothing if the value is not equal.

**90.**Write a Python program to create a copy of its own source code.

**91.**Write a Python program to swap two variables.

**92.**Write a Python program to define a string containing special characters in various forms.

**93.**Write a Python program to get the Identity, Type, and Value of an object.

**94.**Write a Python program to convert the bytes in a given string to a list of integers.

**95.**Write a Python program to check whether a string is numeric.

**96.**Write a Python program to print the current call stack.

**97.**Write a Python program to list the special variables used in the language.

**98.**Write a Python program to get system time.

Note : The system time is important for debugging, network information, random number seeds, or something as simple as program performance.

**99.**Write a Python program to clear the screen or terminal.

**100.**Write a Python program to get the name of the host on which the routine is running.

**101.**Write a Python program to access and print a URL's content to the console.

**102.**Write a Python program to get system command output.

**103.**Write a Python program to extract the filename from a given path.

**104.**Write a Python program to get the effective group id, effective user id, real group id, and a list of supplemental group ids associated with the current process.  
Note: Availability: Unix.

**105.**Write a Python program to get the users environment.

**106.**Write a Python program to divide a path by the extension separator.

**107.**Write a Python program to retrieve file properties.

**108.**Write a Python program to find the path to a file or directory when you encounter a path name.

**109.**Write a Python program to find the path to a file or directory when you encounter a path name.

**110.**Write a Python program to get numbers divisible by fifteen from a list using an anonymous function.

**111.**Write a Python program to make file lists from the current directory using a wildcard.

**112.**Write a Python program to remove the first item from a specified list.

**113.**Write a Python program that inputs a number and generates an error message if it is not a number

**114.**Write a Python program to filter positive numbers from a list.

**115.**Write a Python program to compute the product of a list of integers (without using a for loop).

**116.**Write a Python program to print Unicode characters.

**117.**Write a Python program to prove that two string variables of the same value point to the same memory location

**118.**Write a Python program to create a bytearray from a list.

**119.**Write a Python program to round a floating-point number to a specified number of decimal places.

**120.**Write a Python program to format a specified string and limit the length of a string.

**121.**Write a Python program to determine if a variable is defined or not.

**122.**Write a Python program to empty a variable without destroying it.

Sample data: n=20  
d = {"x":200}  
Expected Output : 0  
{}

**123.**Write a Python program to determine the largest and smallest integers, longs, and floats.

**124.**Write a Python program to check whether multiple variables have the same value.

**125.**Write a Python program to sum all counts in a collection.

**126.**Write a Python program to get the actual module object for a given object.

**127.**Write a Python program to check whether an integer fits in 64 bits.

**128.**Write a Python program to check whether lowercase letters exist in a string.

**129.**Write a Python program to add leading zeroes to a string.

**130.**Write a Python program that uses double quotes to display strings.

**131.**Write a Python program to split a variable length string into variables.

**132.**Write a Python program to list the home directory without an absolute path.

**133.**Write a Python program to calculate the time runs (difference between start and current time) of a program.

**134.**Write a Python program to input two integers on a single line.

**135.**Write a Python program to print a variable without spaces between values.  
Sample value : x =30  
Expected output : Value of x is "30"

**136.**Write a Python program to find files and skip directories in a given directory.

**137.**Write a Python program to extract a single key-value pair from a dictionary into variables.

**138.**Write a Python program to convert true to 1 and false to 0.

**139.**Write a Python program to validate an IP address.

**140.**Write a Python program to convert an integer to binary that keeps leading zeros.  
Sample data : x=12  
Expected output : 00001100  
0000001100

**141.**Write a python program to convert decimal to hexadecimal.  
Sample decimal number: 30, 4  
Expected output: 1e, 04

**142.**Write a Python program to check if every consecutive sequence of zeroes is followed by a consecutive sequence of ones of same length in a given string. Return True/False.  
Original sequence: 01010101  
Check if every consecutive sequence of zeroes is followed by a consecutive sequence of ones in the said string:  
True  
Original sequence: 00  
Check if every consecutive sequence of zeroes is followed by a consecutive sequence of ones in the said string:  
False  
Original sequence: 000111000111  
Check if every consecutive sequence of zeroes is followed by a consecutive sequence of ones in the said string:  
True  
Original sequence: 00011100011  
Check if every consecutive sequence of zeroes is followed by a consecutive sequence of ones in the said string:  
False

**143.**Write a Python program to determine if the Python shell is executing in 32-bit or 64-bit mode on the operating system.

**144.** Write a Python program to check whether a variable is an integer or string.

**145.**Write a Python program to test if a variable is a list, tuple, or set.

**146.**Write a Python program to find the location of Python module sources.

**147.** Write a Python function to check whether a number is divisible by another number. Accept two integer values from the user.

**148.** Write a Python function to find the maximum and minimum numbers from a sequence of numbers.  
Note: Do not use built-in functions.

**149.** Write a Python function that takes a positive integer and returns the sum of the cube of all positive integers smaller than the specified number.

**150.** Write a Python function to check whether a distinct pair of numbers whose product is odd is present in a sequence of integer values.