While looping.	
1.	WAP to print the ‘‘hello world’’ for 5 times by using a while loop.
2.	WAP to print the number from 1 to 10 by using a while loop.
3.	WAP to print the natural number until 1 to nth times by using a while loop.
4.	WAP to print the reverse of the natural number nth to 0 times by using a while loop.
5.	WAP to print the whole number until nth times by using a while loop.
6.	WAP to print the reverse of the whole number until nth times by using a while loop.
7.	WAP to print the even number until nth times by using a while loop.
8.	WAP to print reserve the even number until nth times by using a while loop.
9.	WAP to print the odd number until nth times by using a while loop.
10.	WAP to print reserve the odd number until nth times by using a while loop.
11.	WAP to print the nth table 1 to 20, format ==> 5 * 1 = 5 to 5 * 20 = 100
12.	WAP to print the nth table 20 to 1, format ==> 5 * 20 = 100 to 5 * 1 = 5
13.	WAP to print the numbers which are divisible by 3 from 25 to 66.
14.	WAP to print the numbers which are divisible by 3 and 5 from 15 to 76.
15.	WAP to print the numbers which are divisible by 5 and 7 from 77 to 22.
16.	WAP to print the ’’love you’’ if value is divisible by 3, or print ‘‘hate you’’ value is divisible by 5, to print ‘‘i like you’’ if divisible by 3 and 5, from 1 to 30.
17.	WAP to print sum of the 1 to nth natural numbers.
18.	WAP to print sum of the even numbers 1 to 20 numbers.
19.	WAP to print products of the 1 to 6 numbers.
20.	WAP to count the no of odd values from 1 to 30 numbers.
21.	WAP to print the factors of the given number.
22.	WAP to sum of the factors of the given number.
23.	WAP to prod of the factors of the given number.
24.	**WAP to print Fibonacci series 0 to nth.**
25.	**WAP to find out the factorial number of a given number.**
26.	**WAP to print the factorial numbers in the given range m to n.**

27.	**WAP to check whether a given number is a prime number or not.**
28.	**WAP to print the prime numbers in the given range m to n.**
29.	**WAP to check whether a given number is an armstrong number or not.**
30.	**WAP to print the armstrong numbers in the given range m to n.**
31.	**Write a program to print prime series 1 to nth.**
32.	**Write a program to print 1 to 100 prime numbers in fibonacci series**
33.	**Write a program Sum of prime numbers in a given collection**
34.	WAP to the given number is palindrome or not.
35.	**WAP to check whether the length of the string is even or odd. If it is even, print that string 10 times. If it is not printed 5 times.**
36.	WAP to print the uppercase character from 'A' to 'Z' by using a while loop.
37.	WAP to print the lowercase character from 'a' to 'z' by using a while loop.
38.	WAP to print the ascii numbers character from '0' to '9' by using a while loop.
39.	WAP to print the uppercase character from 'E' to 'U' by using a while loop.
40.	WAP to print the uppercase character from 'Z' to 'A' by using a while loop.
41.	WAP to print the uppercase character from 'U' to 'G' by using a while loop.
42.	WAP to print the lowercase character from 'i' to 'y' by using a while loop.
43.	WAP to print the lowercase character from 'z' to 'a' by using a while loop.
44.	WAP to print the lowercase character from 'p' to 'c' by using a while loop.
45.	WAP to print the uppercase characters based on even ascii values.
46.	WAP to print the lowercase characters based on even ascii values.
47.	WAP to print the uppercase characters based on odd ascii values.
48.	WAP to print the lowercase characters based on odd ascii values.
49.	WAP to print all the even characters from 'A' to 'Z'.
50.	WAP to print all the odd characters from 'a' to 'z'.
51.	WAP to count the number of vowels present in D to V.
52.	WAP to count the number of vowels present in u to e.

53.	**WAP to print the square of the number in between the range from m to n which is multiple of 3. WAP to print a number from range m to m which is divisible by 5.**
54.	**Write a program which takes an integer number and find the sum of digits and repeat until the sum gets a single digit in the end.** **Example:** **Input 5643** **Example:** **5+6+4+3 =>18** **1+8=>9** **Output 9**
55.	**Write a program to count the numbers of 2's between 0 and n** **Input n=35** **Example: 2,12,20,21,22,23,24,25,26,27,28,29,32** **Output 14**
56.	**Write a program to find max product of three numbers from the given integer array** **Input arr() =6,3,2,0,10** **Example: 10*6*2** **Output =180.**
57.	Write a java program to find the largest number 'M' less than a given number 'N' which should not contain a given digit 'D'. For example, if 150 is the given number and 5 is the given digit, then you should find the largest number less than 150 such that it should not contain 5 in it. In this case, 149 will be the answer.
58.	WAP to print all the characters in a given string.
59.	**WAP to print all the items present in the string between the range from m to n.**
60.	WAP to print the uppercase characters in a given string.
61.	WAP to print the lowercase characters in a given string.
62.	WAP to print the ascii numbers(Digits) characters in a given string.
63.	WAP to print the special characters in a given string.
64.	WAP to Print the vowels in a given string.
65.	WAP to Print the consonants in a given string.
66.	WAP to print the reverse of the characters in a given string.
67.	WAP to print even position values characters in a given string.

68.	WAP to print odd position values characters in a given string.
69.	WAP to count the total Number of characters in a given string.
70.	**WAP to count alphabets in a given string.**
71.	WAP to count Number of uppercase in a given string.
72.	**WAP to count lowercase in a given string.**
73.	WAP to count Number of special symbols in a given string.
74.	**WAP to count uppercase and lowercase in a given string.**
75.	WAP to count Number of ascii numbers in a given string.
76.	**WAP to Count Total Number of Words in a String.**
77.	**WAP to Count Total Number of spaces in a String.**
78.	**WAP to Count Vowels in a String.**
79.	**WAP to Count consonants in a String.**
80.	**WAP to Count Vowels and Consonants in a String.**
81.	WAP to count Number of new lines in a given string.
82.	WAP to count Number of lines in a given string.
83.	WAP to count no of alphabets, numbers, and special characters in a given string.
84.	**WAP to Count Total no of words in a first line of the Doc String.**
85.	WAP to sum of digits in a given string.
86.	**WAP to Print First Occurrence of a Character in a String.**
87.	**WAP to Print Last Occurrence of a Character in a String.**
88.	**WAP to Print First not Occurrence of a Character in a String.**
89.	**WAP to Print last not Occurrence of a Character in a String.**
90.	**WAP to Remove First Occurrence of a Character in a String.**
91.	**WAP to Remove Last Occurred Character in a String.**
92.	WAP to count the occurrence of a specific character in a given string.
93.	WAP to count the max occurrence of a character in a given string.
94.	WAP to count the min occurrence of a character in a given string.

95.	WAP to count the repeats of a specific character in a given string.
96.	WAP to count the max repeated character in a given string.
97.	WAP to count the min repeated character in a given string.
98.	WAP to replace a specific character into a new character.
99.	WAP to convert lowercase to uppercase in a given string.
100.	WAP to convert uppercase to lowercase in a given string.
101.	**WAP to convert uppercase to lowercase in even position characters in a given string.**
102.	**WAP to convert uppercase to lowercase in odd position characters in a given string.**
103.	**WAP to convert the uppercase letter to lowercase letter and lowercase letter to uppercase letter, if the character is special symbol replace with * for the given input string.**
104.	WAP to convert title case in a given string.
105.	WAP to convert snake_case in a given string.
106.	**WAP to Toggle Characters Case in a String.**
107.	**WAP to replace vowels into ascii values in a given string.(both with and with temp variable)**
108.	**WAP to replace consonants into ascii values in a given string. (both with and with temp variable)**
109.	**WAP to replace consonants to ascii values in odd position characters in a given string.(both with and with temp variable)**
110.	**WAP to replace vowels to ascii values in even position characters in a given string.(both with and with temp variable)**
111.	**WAP to Replace white Spaces with Hyphen in a String.(both with and with temp variable)**
112.	**WAP to Remove Odd position Characters in a given String.**
113.	**WAP to Remove even position Characters in a given String.**
114.	**WAP to Remove alphabets in a given String.**
115.	**WAP to Remove ascii numbers in a given String.**

116.	**WAP to Remove special characters in a given String.**
117.	**WAP to check if the given string is Palindrome or Not.**
118.	WAP to check whether the given string is lowercase or not.
119.	WAP to check whether the given number is string or not.
120.	I/p: 'hello world 123 haii' o/p:'eoo123aii'
121.	I/p: 'hello world 123 haii' o/p:'eo o 123 aii'
122.	I/p: 'hello world 123 haii' o/p:'eo o aii'
123.	I/p: 'hello world 123 haii' o/p:'hll world haii'
124.	i/p:'hello world' o/p:'hll wrld' o/p:'eoo'
125.	i/p:'12hello3 4world4' o/p:'hll wrld' o/p:'eoo' o/p:'12344'
126.	i/p:'(*12hello3$ $4#world4*)' o/p:'hllwrld' o/p:'eoo' o/p:'12344' o/p:'(*$ $#*)
127.	i/p: '(*12hello3$ $4#world4*)' o/p: ')*#$ $*(,44321,ooe,dlrwllh'
128.	i/p:'(*12hello3$ $4#world4*)' o/p:'hllwrld,eoo,12344,(*$ $#*)'
129.	i/p: '(*12hello3$ $4#world4*)' o/p:',hllwrld,)*#$ $*(',eoo,44321'
130.	i/p: '(*12hello3$ $4#world4*)' o/p: ')*#$ $*(,44321,ooe,dlrwllh'
131.	i/p: '(*12hello3$ $4#world4*)' o/p:'dlrwllh)*#$ $*(ooe44321'
132.	I/p:'HELLOWORLD' o/p:'h$ll*w*rld'

133.	#I/p:'123HELLO WORLD456' #o/p:'hllwrld21'
134.	#I/p:'123HELLO WORLD456' #o/p:'123hEllO wOrld456'
135.	#I/p:'123HELLO WORLD456' #o/p:'321hEllO wOrld654'
136.	#I/p:'123HELLO WORLD456' #o/p:'321*E**O *O***654'
137.	#I/p:'123HELLO WORLD456' #o/p:'123*E**O *O***456'
138.	#I/p:'HELLOWORLD' #o/p:'h$ll*w*rld'
139.	i/p: 'abc' i/p: 'xyz' o/p:'axbycz'
140.	i/p: 'abc' i/p: 'xyzw' o/p:'axbyczw'
141.	i/p: 'abcd' i/p: 'xyz' o/p:'axbyczd'
142.	i/p: 'abc' i/p: 'xyzwop' o/p:'axbyczwop'
143.	i/p:'abc123' o/p:'a1b2c3'
144.	i/p:'abc12' o/p:'a1b2c'
145.	i/p: 'abc1234' o/p: 'a1b2c34'
146.	i/p: '1a2bc34' o/p:'a1b2c34'
147.	i/p:'abcdnmpoijkl12yzabcd5'

	o/p:3
148.	I/P: ‘HAII HELLO’ O/P: ‘HA1I2I3 HE4LLO5’
149.	I/P: ‘HAII HELLO’ O/P: ‘H1AII 2H3EL4L5O’
150.	**WAP to print all the string values present inside the list if the length of the string is odd under the first character starts with a vowel.**
151.	**Write a program for series Input:- “a3b2c4” , Output: “aaabbcccc”**
152.	**Write a program for series Input: “aaabbcccc” Output:- “a3b2c4”**
153.	**Write a program frequency of each character in given string when count is a prime number input:”aaaaBBBcddddfffff” Output: (“B3”,”c1”,”f5”)**
154.	**Write a program to swap neighbor char in string. Input: “abcdefg” Output:- “badcfeg”**
155.	**WAP to split the sentence in a given string. i/p:’haii how are you venu’ o/p:[‘haii’, ’how’,’are’,’you’,’venu’]**
156.	**Find out the longest word in the string below. Sentence =”hello world welcome to python”**
157.	**sentence=” hello world welcome to python” Output: ’olleh dlrow emoclew ot nohtyp’**
158.	**sentence=” python is a programming language” (Using dictionary comprehension) Output: {‘python’:1, is:1,’a’:1,’language’:1,’programming:1’}**
159.	I/p:'PYTHON SCRIPTING PROGRAMMING LANGUAGE' o/p:['PYTHON’,'SCRIPTING’, ‘PROGRAMMING’,’ LANGUAGE']
160.	I/p:'PYTHON SCRIPTING PROGRAMMING LANGUAGE' o/p:['nothyp', 'scripting','gnimmargorp', 'language']

161.	**I/p: 'ABCDEFGIJKABC'** **O/p: ['ABCDEFG', 'IJK', ' ABC']**
162.	WAP to check whether two given strings contain the same set of characters but in different order.For example, "POT" and "OTP" are having the same set of characters.
163.	WAPto reverse alternate words of a given string starting with the first word and the swap first and last char of the remaining words. For example, if "Java Concept for The Day" is input string then output should be "avaJ toncepc rof eht yaD"
164.	Find the sum of all numbers present in the string and consider -12 as a negative number in string. Ex: saf67as71mha-12v1 o/p: 67 + 71 - 12 + 1 = 127
165.	WAP You are given a string and a number n. Remove the characters which repeat n times or more consecutively and print the remaining string. Ex: n=2 Input: Shaanvi Output: Shnvi
166.	WAP Two strings, and are called anagrams if they contain all the same characters in the same frequencies. For example, the anagrams of CAT are CAT, ACT, TAC, TCA, ATC, and CTA. If and are case-insensitive anagrams, print "Anagrams"; otherwise, print "Not Anagrams" instead. Constraints o Length of both strings in between 1 - 50. o Strings consist of English alphabetic characters. o The comparison should NOT be case sensitive. Sample Input anagram margana Sample Output Anagrams
167.	WAP to find the longest substring without repeating characters in a given string. For example, if "javaconceptoftheday" is the input string, then the longest substring without repeating or duplicate characters is "oftheday".
168.	WAP to convert the given string lower case to upper and upper case to lower. For example,

	Sample Input: "WelcOMe to YuPP videO serVicES" Sample Output: wElComE TO yUpp VIDEo SERvICes
169.	Write a program to remove the duplicates in a given list. For example, Sample Input: [5, 3, 5, 2, 1, 6, 6, 4] Sample Output: [5, 3, 2, 1, 6, 4]
170.	WAP to reverse words in a given sentence without using any library method. Sample Input: Yupp video services Sample Output: ppuy oediv secivres
171.	Given two words, check if both are formed with the same distinct characters. Ex: sairam, sarmi
172.	WAP to print the data items in a given list.
173.	WAP to reverse the data items in a given list.
174.	WAP to extract and store the single value data types in a given list. (store the values in list)
175.	WAP to extract and store the multi value data types in a given list. (store the values in tuple)
176.	WAP to extract and store the mutable values data types in a given list. (store the values in set)
177.	WAP to extract and store the immutable value data types in a given list. (store the values in tuple)
178.	**WAP to print only string items present in the list.**
179.	**WAP To Calculate the Average of Numbers in a Given List.**
180.	**WAP To Accept Three Digits and Print all Possible Combinations from the Digits**
181.	**WAP to print only string items present in the list in between the range from m to n.**
182.	WAP to extract and store the string value in a given list. (store the values in list)

183.	**WAP to print all the even numbers present in the odd index position inside the tuple.**
184.	**WAP to print all the odd numbers present in the even index position inside the list.**
185.	**WAP to count the number of duplicate elements in a given list or tuple.**
186.	**WAP Sum of duplicate elements(single value data types) in a given collection**
187.	**WAP to find out the first max value in a given collection.**
188.	**WAP to find out the second max value in a given collection.**
189.	**WAP to find out the first min value in a given collection.**
190.	**WAP to find out the second max value in a given collection.**
191.	wap count the number of sub same collections in a main collection. ip= [1,2,3,[1,2],[2,5]] op:2
192.	Wap to count the number of even and odd numbers is given a list. i/p:[1,8,9,7,25,26,98,75,74,36,15] o/p:{'even':5,'odd':6}
193.	**I/P:** **List_1 = [1, 2, 5, ”HAI, ”Hello”, 4+10j],** **List_2= [3, ”GOOD”, “HAVE”, ”A”, ”NICE”, ”DAY”]** **o/p:** **List_1 = [1, 2, 5, 3, 4+10j],** **List_2= [”HAI, ”Hello”, ”GOOD”, “HAVE”, ”A”, ”NICE”, ”DAY”]**
194.	**Input=[10,10,10,20,30,40,40,50,60,70,70,70]** **o/p:[10,20,30,40,50,60,70]** **Eliminate repeated values.** **Without using built in functions, set, set compershive, membership operator.**
195.	**You are given an array of n numbers and values of l and m. Print the positive elements of the array which are either divisible by l or by m but not by both. Consider the number if divisible by both and the sum of digits is 6.** **Ex: arr = {2,3,5,6,8,12,10,60,66} l = 2, m = 3** **O/P: {2,3,6,8,10,60}**
196.	**Write a program to find out the 2nd largest number in an array.** **For example: [1,2,3,4,5,6,7,8,9,0].**

197.	WAP to find a continuous subarray of the given array whose sum is equal to a given number. For example, If {52, 10, 32, 9, 21, 8} is the given array and 51 is the given number, then you have to find a continuous subarray in this array such that its elements add up to 51. In this case, {10, 32, 9} is such a sub array whose elements add up to 51.
198.	WAP Arrange the elements of the number array such that all negative numbers are at the left. Print one possible solution. Ex: {-5, 4, -9, 1, 0} Op: {-5, -9, 4, 1, 0} {-5, -9, 1, 4, 0}
199.	wap li={'google.com','facebook.in','python.edu','youtube.com'} op:{'6':['google','python'],'7':['youtube'],'8':['facebook']}
200.	wap li={'google.com','facebook.in','python.edu','youtube.com'} op=[['com',2],['in',1],['edu',1]]
201.	wap st='welcome to pyspiders all the best your career ' op=('reerac','ruoy','tseb','eth','lla','sredipsys','ot','emoclew')
202.	wap st='welcome to pyspiders all the best your career' op:('welcome','ot','pyspiders','lla','the','tesb','your','reerac')
203.	wap li=('A','C','Z','a','c','d') op:{'65':1,'67':2,'90':6,'97':24,'99':120,'100':720}
204.	**mylist = [[1,"abc",'2012-07-01'],[1,"dsc",'2012-07-02'],[2,"abc",'2012-07-01'],[2,"dsc",'2012-07-02'],[3,"asdc",'2012-07-04'],[2,"sasa",'2012-07-05']]** **output= [[1,"abc",'2012-07-01',"dsc",'2012-07-02'],[2,"abc",'2012-07-01',"dsc",'2012-07-02',"sasa",'2012-07-05'],[3,"asdc",'2012-07-04']]**
205.	**a=[1, 'planA', '10000.25|10000.50|10000.15']** **o/p:{'planA':30000, 'avgvalue':10000.0}**