|  |
| --- |
| Question 1: |
|  |

|  |
| --- |
| Write a program that calculates and prints the value according to the given formula: |
|  |

|  |
| --- |
| Q = Square root of [(2 \* C \* D)/H] |
|  |

|  |
| --- |
| Following are the fixed values of C and H: |
|  |

|  |
| --- |
| C is 50. H is 30. |
|  |

|  |
| --- |
| D is the variable whose values should be input to your program in a comma-separated sequence. |
|  |

|  |
| --- |
| Example |
|  |

|  |
| --- |
| Let us assume the following comma separated input sequence is given to the program: |
|  |

|  |
| --- |
| 100,150,180 |
|  |

|  |
| --- |
| The output of the program should be: |
|  |

18,22,24

Ans: ##calculates and prints the value according to the given formula,Q = Square root of [(2 \* C \* D)/H]  
#values of C & H  
print("Values of C and H are 50 and 30 respectively")  
C=50  
H=30  
l=int(input("enter the no of outputs you want"))  
print("enter the values of D ")  
List=[]  
output=[]  
for i in range(l):  
 D=int(input())  
 List.append(D)  
print("the entered inputs are: ")  
for i in List:  
 print(i,end=',')  
import math  
for i in List:  
 Q = math.sqrt((2 \* C \* i)/H)  
 output.append(Q)  
print("\n")  
print("the corresponding output are: \n")  
for i in output:  
 print(i,end=',')

|  |
| --- |
| Question 2: |
| Write a program which takes 2 digits, X,Y as input and generates a 2-dimensional array. The element value in the i-th row and j-th column of the array should be i\*j. | |
|  | |

|  |
| --- |
| Note: i=0,1.., X-1; j=0,1,¡­Y-1. |
|  |

|  |
| --- |
| Example |
|  |

|  |
| --- |
| Suppose the following inputs are given to the program: |
|  |

|  |
| --- |
| 3,5 |
|  |

|  |
| --- |
| Then, the output of the program should be: |
|  |

|  |
| --- |
| [[0, 0, 0, 0, 0], [0, 1, 2, 3, 4], [0, 2, 4, 6, 8]] |
| Ans: |
|  |

row\_num = int(input("Input number of rows: "))  
col\_num = int(input("Input number of columns: "))  
multi\_list = [[0 for col in range(col\_num)] for row in range(row\_num)]  
  
for row in range(row\_num):  
 for col in range(col\_num):  
 multi\_list[row][col]= row\*col  
  
print(multi\_list)

Question 3:

|  |
| --- |
| Write a program that accepts a comma separated sequence of words as input and prints the words in a comma-separated sequence after sorting them alphabetically. |
|  |

|  |
| --- |
| Suppose the following input is supplied to the program: |
|  |

|  |
| --- |
| without,hello,bag,world |
|  |

|  |
| --- |
| Then, the output should be: |
|  |

bag,hello,without,world

Ans: ##accepts a comma separated sequence of words as input  
##and prints the words in a comma-separated sequence after sorting them alphabetically.  
#inputs  
word=[]  
l=int(input("enter the no of words to be inserted :"))  
print("enter words :")  
for i in range(l):  
 w=input()  
 word.append(w)  
print("the entered words are :")  
for i in word:  
 print(i,end=',')  
# sorting words  
output=word  
for i in range(l):  
 for j in range(l-1):  
 if output[j]>output[j+1]:  
 output[j],output[j+1]=output[j+1],output[j]  
#output  
print("\n")  
print("the sorted words :")  
for i in output:  
 print(i,end=',')

Question 4:

|  |
| --- |
| Write a program that accepts a sequence of whitespace separated words as input and prints the words after removing all duplicate words and sorting them alphanumerically. |
|  |

|  |
| --- |
| Suppose the following input is supplied to the program: |
|  |

|  |
| --- |
| hello world and practice makes perfect and hello world again |
|  |

|  |
| --- |
| Then, the output should be: |
|  |

again and hello makes perfect practice world

##sorting them alphanumerically.  
#inputs  
str=input("enter a string :")  
word=str.split()  
word\_new=[]  
word\_noduplicate=[]  
#removing duplicates  
for i in word:  
 if i not in word\_new:  
 word\_new.append(i)  
 word\_noduplicate.append(i)  
#sorting  
word\_sorted=word\_new  
for i in range(len(word\_sorted)):  
 for j in range(len(word\_sorted)-1):  
 if word\_sorted[j]>word\_sorted[j+1]:  
 word\_sorted[j],word\_sorted[j+1]=word\_sorted[j+1],word\_sorted[j]  
  
s1=''  
s2=''  
#conversion to string  
for i in range(len(word\_sorted)):  
 s1+=word\_noduplicate[i]+' '  
 s2+=word\_sorted[i]+' '  
print("the words with no duplicates : ",s1)  
print("the sorted word: ",s2)

Question 5:

|  |
| --- |
| Write a program that accepts a sentence and calculate the number of letters and digits. |
|  |

|  |
| --- |
| Suppose the following input is supplied to the program: |
|  |

|  |
| --- |
| hello world! 123 |
|  |

|  |
| --- |
| Then, the output should be: |
|  |

|  |
| --- |
| LETTERS 10 |
|  |

DIGITS 3

Ans: ##accepts a sentence and calculate the number of letters and digits  
#input  
str=input("enter the sentence : ")  
#storing ascii of alphabets and digits  
ascii\_alphabet=[]  
for i in range(ord('A'),ord('Z')):  
 ascii\_alphabet.append(i)  
for i in range(ord('a'),ord('z')):  
 ascii\_alphabet.append(i)  
ascii\_digit=[]  
for i in range(ord('0'),ord('9')):  
 ascii\_digit.append(i)  
#counting no of alphabets and digits  
c=0  
d=0  
for i in str:  
 for j in ascii\_alphabet:  
 if ord(i)==j:  
 c+=1  
 for k in ascii\_digit:  
 if ord(i)==k:  
 d+=1  
print("the no of alphabets is {} and no of digits is {} ".format(c,d))

Question 6:

|  |
| --- |
| A website requires the users to input username and password to register. Write a program to check the validity of password input by users. |
|  |

|  |
| --- |
| Following are the criteria for checking the password: |
|  |

|  |
| --- |
| 1. At least 1 letter between [a-z] |
|  |

|  |
| --- |
| 2. At least 1 number between [0-9] |
|  |

|  |
| --- |
| 1. At least 1 letter between [A-Z] |
|  |

|  |
| --- |
| 3. At least 1 character from [$#@] |
|  |

|  |
| --- |
| 4. Minimum length of transaction password: 6 |
|  |

|  |
| --- |
| 5. Maximum length of transaction password: 12 |
|  |

|  |
| --- |
| Your program should accept a sequence of comma separated passwords and will check them according to the above criteria. Passwords that match the criteria are to be printed, each separated by a comma. |
|  |

|  |
| --- |
| Example |
|  |

|  |
| --- |
| If the following passwords are given as input to the program: |
|  |

|  |
| --- |
| ABd1234@1,a F1#,2w3E\*,2We3345 |
|  |

|  |
| --- |
| Then, the output of the program should be: |
|  |

ABd1234@1

Ans:

##to check the validity of password input by users.  
#input  
k=int(input("the no of passwords : "))  
print("enter password")  
temp=[]  
for i in range(k):  
 password=input()  
 temp.append(password)  
print("the entered passwords are : ")  
for i in temp:  
 print(i,end=',')  
#storing ascii of alphabets and digits  
ascii\_alphabet=[]  
for i in range(ord('A'),ord('Z')):  
 ascii\_alphabet.append(i)  
for i in range(ord('a'),ord('z')):  
 ascii\_alphabet.append(i)  
ascii\_digit=[]  
for i in range(ord('0'),ord('9')):  
 ascii\_digit.append(i)  
ascii\_S=[]  
S=['@','#','$']  
for i in S:  
 ascii\_S.append(ord(i))  
asc=ascii\_alphabet+ascii\_digit+ascii\_S  
#checking for password  
  
pw\_check=[]  
print('\n')  
for i in temp:  
 pw\_s = ''  
 lis=list(i)  
 p = []  
 for j in lis:  
 for l in asc:  
 if ord(j)==l:  
 p.append(j)  
 if len(p)>5 and len(p)<13:  
 for i in p:  
 pw\_s+=i  
 pw\_check.append(pw\_s)  
print("the correct passwords entered : ")  
for i in pw\_check:  
 print(i,end=',')  
print('\n',"the other passwords if not printed could be incorrect due to non-match to following criteria")  
print("1. At least 1 letter between [a-z]")  
print("2. At least 1 number between [0-9]")  
print("3. At least 1 letter between [A-Z]")  
print("4. At least 1 character from [$#@]")  
print("5. Minimum length of transaction password: 6 ")  
print("6. Maximum length of transaction password: 12")