| 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

Sol:-

import math

# Fixed values of C and H

C = 50

H = 30

# Input sequence of D values

input\_sequence = input("Enter comma-separated sequence of D values: ")

# Split the input sequence by comma and convert to list of integers

D\_values = [int(i) for i in input\_sequence.split(',')]

# Calculate Q value for each D value and append to results list

results = []

for D in D\_values:

Q = math.sqrt((2 \* C \* D) / H)

results.append(round(Q))

# Print the results

print(','.join(map(str, results)))

| 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]]  Sol:-  # Get input values of X and Y  X, Y = map(int, input("Enter two digits separated by comma: ").split(','))  # Initialize a 2D array with zeros  array = [[0 for j in range(Y)] for i in range(X)]  # Fill the array with values  for i in range(X):  for j in range(Y):  array[i][j] = i \* j  # Print the 2D array  print(array) |
| --- |
|  |

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

Sol:-

# Get input sequence of words

words = input("Enter comma separated words: ").split(',')

# Sort the words alphabetically

words.sort()

# Join the words into a comma-separated string

result = ','.join(words)

# Print the result

print(result)

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

Sol:-

# Get input sequence of whitespace separated words

words = input("Enter whitespace separated words: ").split()

# Remove duplicates and sort the words alphanumerically

unique\_words = sorted(set(words))

# Join the unique words into a whitespace-separated string

result = ' '.join(unique\_words)

# Print the result

print(result)

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

Sol:-

sentence = input("Enter a sentence: ")

letter\_count = 0

digit\_count = 0

for char in sentence:

if char.isalpha():

letter\_count += 1

elif char.isdigit():

digit\_count += 1

print("LETTERS", letter\_count)

print("DIGITS", digit\_count)

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

Sol:-

import re

def check\_password(password):

if len(password) < 6 or len(password) > 12:

return False

if not re.search("[a-z]", password):

return False

if not re.search("[0-9]", password):

return False

if not re.search("[A-Z]", password):

return False

if not re.search("[$#@]", password):

return False

return True

passwords = input("Enter comma-separated passwords: ")

password\_list = passwords.split(",")

valid\_passwords = []

for password in password\_list:

if check\_password(password):

valid\_passwords.append(password)

print(",".join(valid\_passwords))