|  |
| --- |
| 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

import math

# Fixed values

C = 50

H = 30

# Input sequence of D values

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

# Split the input sequence into individual values

values = input\_sequence.split(',')

# Calculate Q for each value of D and store the results in a list

results = []

for d in values:

d = int(d) # Convert the string value to an integer

q = math.sqrt((2 \* C \* d) / H)

results.append(int(q))

# Convert the results list to a comma-separated string

output = ','.join(str(q) for q in results)

# Print the output

print(output)

|  |
| --- |
| 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]] |
| # Input X and Y  X, Y = map(int, input("Enter two digits (X, Y): ").split(','))  # Generate the 2-dimensional array  array = [[i \* j for j in range(Y)] for i in range(X)]  # Print the array  for row in array:  print(row) |

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

# Input comma-separated sequence of words

words = input("Enter a comma-separated sequence of words: ").split(',')

# Sort the words alphabetically

sorted\_words = sorted(words)

# Join the sorted words into a comma-separated sequence

result = ','.join(sorted\_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

# Input sequence of whitespace-separated words

words = input("Enter a sequence of whitespace-separated words: ").split()

# Remove duplicate words using set

unique\_words = set(words)

# Sort the unique words alphanumerically

sorted\_words = sorted(unique\_words)

# Join the sorted words into a whitespace-separated string

result = ' '.join(sorted\_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

# Input sentence

sentence = input("Enter a sentence: ")

# Initialize counters for letters and digits

letter\_count = 0

digit\_count = 0

# Iterate over each character in the sentence

for char in sentence:

if char.isalpha():

# Increment letter count if the character is a letter

letter\_count += 1

elif char.isdigit():

# Increment digit count if the character is a digit

digit\_count += 1

# Print the results

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

import re

# Passwords input

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

# Split the passwords by comma

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

# Regular expression pattern for password validation

pattern = r"^(?=.\*[a-z])(?=.\*[A-Z])(?=.\*\d)(?=.\*[$#@]).{6,12}$"

# Iterate over each password

valid\_passwords = []

for password in password\_list:

if re.match(pattern, password):

valid\_passwords.append(password)

# Print the valid passwords

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