Python Programming Basic Assignment 13

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

In [7]:

```python
import math
# Fixed values of C and H
C = 50
H = 30
# Read the input values of D as a comma-separated string
D_input = '100,150,180'
# Split the string into a list of integers
D_values = [int(x) for x in D_input.split(',')]
# Calculate and print the value of Q for each value of D
for D in D_values:
    Q = int(math.sqrt((2 * C * D) / H))
    print(Q, end=",")
```

```
18,22,24,
```

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.

In [11]:

```python
# Read the input values X and Y
X, Y = map(int, input().split())
# Create an empty 2-dimensional array
array = []
# Generate the array using a nested loop
for i in range(X):
    row = []
    for j in range(Y):
        row.append(i * j)
    array.append(row)
# Print the array
print(array)
```

```
4 6
[[0, 0, 0, 0, 0, 0], [0, 1, 2, 3, 4, 5], [0, 2, 4, 6, 8, 10], [0, 3, 6, 9, 12, 15]]
```

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.

In [12]:

```
# Read the input string
words = 'hello world hello goodbye world'
# Split the string into a list of words
word_list = words.split(',')
# Sort the list of words alphabetically
sorted_words = sorted(word_list)
# Join the sorted list of words into a single string
output_string = ','.join(sorted_words)
# Print the output string
print(output_string)
```

```
hello world hello goodbye world
```

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.

In [13]:

```
# Read the input string
words = 'hello world hello goodbye world'
# Split the string into a list of words
word_list = words.split()
# Remove the duplicate words from the list
unique_words = set(word_list)
# Sort the list of words alphabetically
sorted_words = sorted(unique_words)
# Join the sorted list of words into a single string
output_string = ' '.join(sorted_words)
# Print the output string
print(output_string)
```

```
goodbye hello world
```

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

In [14]:

```
# Read the input string
sentence = 'Hello world! 123'
# Initialize the counters for letters and digits
num_letters = 0
num_digits = 0
# Loop through each character in the string
for ch in sentence:
    # Check if the character is a letter
    if ch.isalpha():
        num_letters += 1
    # Check if the character is a digit
    elif ch.isdigit():
        num_digits += 1
# Print the results
print("LETTERS:", num_letters)
print("DIGITS:", num_digits)
```

```
LETTERS: 10
DIGITS: 3
```

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.

In [16]:

```
# Read the password from the user
password = 'HelloWorld123'
# Set the minimum length of the password
min_length = 8
# Initialize the counters for lowercase letters, uppercase letters, and digits
num_lower = 0
num_upper = 0
num_digits = 0
# Loop through each character in the password
for ch in password:
    # Check if the character is a lowercase letter
    if ch.islower():
        num_lower += 1
    # Check if the character is an uppercase letter
    elif ch.isupper():
        num_upper += 1
    # Check if the character is a digit
    elif ch.isdigit():
        num_digits += 1
   # Check if the password is at least 8 characters long
if len(password) < min_length:
    print("Invalid password")
# Check if the password contains at least one lowercase letter, one uppercase letter,
elif num_lower == 0 or num_upper == 0 or num_digits == 0:
    print("Invalid password")
else:
    print("Valid password")
```

```
Valid password
```

In [ ]: