1. Solomon says:
   Ask the user to input any sentence. Then call the module **solomon_says.py**. In this module is a function **`solomon_add()`** that adds the phrase "Solomon says, (user sentence)" and returns the new sentence. Print this sentence.
   Example.
   Input: "Give me an apple."
   Output: "Solomon says, give me an apple."

2. Take a random string from the user. Calculate and print separately, the number of letters, spaces and digits in that string. To do this create a separate **module file requiredFunctions.py**. In this module file, there should be three separate functions `calcSpaces(), calcDigits()` and `calcLetters()`. Call this module into your main code.
   Example.
   Input - Hi, I am Aditya. I'm an elf and my real age is 300 years.
   Output -
   Number of letters - 37
   Number of digits - 3
   Number of spaces - 13

3. Write a **function** that accepts a hyphen-separated sequence of words and then outputs/prints the same hyphen separated sequence but with all the words sorted alphabetically. Take input from the user.
   Eg.
   Input: boy-apple-giraffe-chipotle-stroopwafel
   Output: apple-boy-chipotle-giraffe-stroopwafel

4. With a given integral number n, write a program to generate a dictionary that contains (i, i*i) such that is an integral number between 1 and n (both included). and then the program should print the dictionary.
   Suppose the following input is supplied to the program:
   7
   Then, the output should be:
   {1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49}
   (the main part of the code should be inside a function. So take input from the user, and then pass it to a function called generateSquares().)

5. **Number Guessing Game:**
   Write a program that accepts a number from the user (any integer between 1 and 10). Your program should also generate a random number. Between 1 and 10. If the number the user entered is the same as the one your program generated the user wins. If not, the user gets three tries, and if unsuccessful, the user loses.
   Hint: check out the **randint()** function in Python3
   Python documentation: https://docs.python.org/3.1/library/random.html

6. Write a program which takes 2 digits (on same line), 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]]
(the main part of the code should be inside a function. So take input from the user, and then pass it to a function called `generateArray()`.)

7. Write a program that computes the value of a+aa+aaa+aaaa with a given digit as the value of a.
Example: Suppose the following input is given:
5
Then, the output should be: 6170
(5 + 55 + 555 + 5555)
(the main part of the code should be inside a function. So take input from the user, and then pass it to a function called `calculateAValue()`.)

8. With two given lists [1,3,6,78,35,55,57] and [12,24,35,24,78,120,155,3], write a program to make a list whose elements are the common elements from the two given lists.
Output:
[3,78,35]
(the main part of the code should be inside a function. Pass the two lists to a function called `commonElements()`.)

9. **Finding a song in a database**

Part1: build database

First write a function that builds a database by taking 3 inputs from the user: `artist, albumName` and `yearOfRelease`. User will supply these 3 input for 5 different artists. So each of the 5 artists you will have the `albumName` and the `yearOfRelease`. You store all 5 sets of input (each set has 3 values) in your program. How you store it is upto you. But of course your storage method should be intelligent (don't use a separate variable for each input). Each input the user gives will be in one seperate line.

Example:

```
Give the details for artist 1: Linkin Park, Meteora, 2003
Give the details for artist 2: Eminem, Revival, 2017
```

...And so on…. 5 times for 5 different artists.

Part2: find the information

Now after collecting the information of 5 artists and storing it, ask the user for an album name. User will enter the album name. If the album name exists in your database, then output the Artist name and the Year of Release for that corresponding Album.

Example: Input:

```
Find information for which album: Meteora.
```

Output:

```
"The Artist who created this album is Linkin Park. They released it in 2003."
```

If the Album name doesn't exist in your database, output the following.

```
"Error Statement : "The album Falling Stars wasn't found in our database. Please check your entry
again."
```

Check out the required program structure on the next slide.

Structure for question 9

Module ... databaseFunctions.py

Functions in this module

-storeNewEntry

-findEntry

-entryNotFound

Main Program

Import both modules in the main program.

Function to take input - takeUserInput

Call the two module functions when needed.

Total:

No of module .py files: 1

No of main program .py files: 1