Password manager

**Do not use** **your real accounts and passwords for this activity.**

# Code snippets .

Creates a 2D list and then changes one item in that list.

| 1  2  3  4  5 | animals = [["Salmon", "Pollock", "Cod"],  ["Parrot", "Duck", "Wren"],  ["Camel", "Lion", "Tiger"]]    animals[0][2] = "Plaice" |
| --- | --- |

Creates a 2D list and appends an item to the list at location 0.

| 1  2  3  4  5  6  7 | animals = [["Salmon", "Pollock", "Cod"],  ["Parrot", "Duck", "Wren"],  ["Camel", "Lion", "Tiger"]]    animals[0].append("Trout")  print(animals) |
| --- | --- |

Creates a list and continuously appends to the list until N is entered.

| 1  2  3  4  5  6  7  8  9  10  11  12 | words = []  words\_left = True  while words\_left:  print("Enter a word")  word = input()  words.append(word)  print("Would you like to enter another word? Y/N")  answer = input().upper()  if answer == "N":  words\_left = False  print(words) |
| --- | --- |

Finds the index location of an entered item.

| 1  2  3 | print("Which word would you like to find?")  word = input()  location = (words[0].index(word)) |
| --- | --- |

# Task . Populate the 2D list

**Step 1**

Use the code below as your starting point for the 2D passwords list. This creates a 2D list that holds three empty lists. These will be populated in the next step.

| 1  2  3 | passwords = [[],  [],  []] |
| --- | --- |

**Step 2**

The incomplete code below should add the item RaspberryPiOS to the first list in the 2D list. Complete it and test it out by printing the list underneath.

|  | passwords ("Raspbian") |
| --- | --- |

**Step 3**

Write two more lines of code that will:

* Add the username pi to the list at location 1
* Add the password raspberry to the list at location 2

**Step 4**

Test your code by printing the passwords list and checking if each list now has a new item. It should look like the output below:

[['RaspberryPiOS'], ['pi'], ['raspberry']]

# Task . Iteratively populate the 2D list

**Step 1**

Incorporate your working code into a while loop. Your program should:

* Continue to ask the user if they would like to enter a new password
* If they say yes, it should prompt for the account, username, and password
* If they say no, then the loop should terminate
* After the loop has terminated, it should print the entire 2D list

**Tip**: Use the code snippet on the first page to help you with this.

# Task . Access an account username and password

**Step 1**

Incorporate extra functionality to your program. After the user has entered all of their passwords, it should no longer print them at the end.

It should:

* Ask the user which account they would like to reveal the password for
* Allow the user to enter the account name
* Search for the account name in the account list (list 0)
* Reveal the username and password for that account based on the found location

| **Example** |  |
| --- | --- |
| **Note:** Given the input you see in this sample interaction, this is the output your program should produce. | |
| The user is given a prompt. | Enter the account |
| The user enters the account name. | raspberrypios |
| The user is given a prompt. | Enter the username |
| The user enters the username. | pi |
| The user is given a prompt. | Enter the password |
| The user enters the password. | raspberry |
| The user is given a prompt. | Would you like to enter another account? Y/N |
| The user enters a response. | n |
| The response was n so the loop terminates and gives the user a prompt. | Which account would you like to see the password for? |
| The user enters the account they wish to view. | raspberrypios |
| The program searches for the account and finds the index location. This is then used to provide output from the other lists. | Account: raspberrypios  Username: pi  Password: raspberry |

# Explorer task .

Improve your program by introducing a menu system and subroutines.

The program should:

* Ask the user if they wish to enter a new account or display the password for an account
* Depending on their choice, call the corresponding subroutine

**Hint:** You need to think carefully about where the list should be held and how other subroutines will access that list.

Your Code:

|  |
| --- |