**Activities: Python Data Structures**

This page contains activities designed to help you practice the skills you learned in the previous lesson. By completing these activities, you will demonstrate your ability to:

* Use lists and tuples to store data collections and retrieve specific values from them.
* Update values in data structures that allow it.

Complete each activity as described and run the code to make sure it works as expected before starting the next activity.

**Activity 1**

Prompt the user to answer a series of 3–5 questions about themselves (such as their name, their age, their birthday, or where they live) and save the answers in a [list](https://academy.engagelms.com/mod/page/view.php?id=185968). Display the results to the user.

|  |
| --- |
| # List to store the user answers  user\_info = []  # Collects the answers from the user  name = input ("what is your name?")  age = input ("How old are you?")  birthday = input ("when is your birthday? Ex. YYYY-MM-DD")  hobby = input ("What is your favorite hobby?")  # Store the answers in the list  user\_info.append(name)  user\_info.append(age)  user\_info.append(birthday)  user\_info.append(hobby)  # Display the results  print("Here are the details you provided:")  print("Name:", user\_info[0])  print("Age:", user\_info[1])  print("Birthday:", user\_info[2])  print("Favorite Hobby:", user\_info[3])  print(user\_info) |

**Activity 2**

1. Present the user with an existing [list](https://academy.engagelms.com/mod/page/view.php?id=185968) of items (such as the [list](https://academy.engagelms.com/mod/page/view.php?id=185968) created in the previous activity).
2. Prompt the user for 2–4 more values to add to the [list](https://academy.engagelms.com/mod/page/view.php?id=185968).
3. Update the [list](https://academy.engagelms.com/mod/page/view.php?id=185968) with the new values.
4. Display the updated [list](https://academy.engagelms.com/mod/page/view.php?id=185968).

|  |
| --- |
| # List to store the user answers  user\_info = []  # Collects the answers from the user  name = input ("what is your name?")  age = input ("How old are you?")  birthday = input ("when is your birthday? Ex. YYYY-MM-DD")  hobby = input ("What is your favorite hobby?")    # Store the answers in the list  user\_info.append(name)  user\_info.append(age)  user\_info.append(birthday)  user\_info.append(hobby)  # Display the results  print("Here are the details you provided:")  print("Name:", user\_info[0])  print("Age:", user\_info[1])  print("Birthday:", user\_info[2])  print("Favorite Hobby:", user\_info[3])  print(user\_info)  # Prompt the user for 2–4 more values to add to the list  value1 = input("Please provide another piece of information about yourself: ")  value2 = input("Please provide one more piece of information about yourself: ")  # Add the new values to the list  user\_info.extend([value1, value2])  # Display the updated list  print("The updated list of your details is:", user\_info) |

**Activity 3**

1. Present the user with a [list](https://academy.engagelms.com/mod/page/view.php?id=185968) of 7–9 items (such as the [list](https://academy.engagelms.com/mod/page/view.php?id=185968) created in the previous activities).
2. Prompt them to enter a value to delete from the [list](https://academy.engagelms.com/mod/page/view.php?id=185968) by value, not by index.
3. Delete the value from the [list](https://academy.engagelms.com/mod/page/view.php?id=185968).
4. Display the updated [list](https://academy.engagelms.com/mod/page/view.php?id=185968).

|  |
| --- |
| # List to store the user answers  user\_info = []  # Collects the answers from the user  name = input ("what is your name?")  age = input ("How old are you?")  birthday = input ("when is your birthday? Ex. YYYY-MM-DD")  hobby = input ("What is your favorite hobby?")  # Store the answers in the list  user\_info.append(name)  user\_info.append(age)  user\_info.append(birthday)  user\_info.append(hobby)  # Display the results  print("Here are the details you provided:")  print("Name:", user\_info[0])  print("Age:", user\_info[1])  print("Birthday:", user\_info[2])  print("Favorite Hobby:", user\_info[3])  print(user\_info)  # Prompt the user for 2–4 more values to add to the list  value1 = input("Please provide another piece of information about yourself: ")  value2 = input("Please provide one more piece of information about yourself: ")  # Add the new values to the list  user\_info.extend([value1, value2])  # Display the updated list  print("The updated list of your details is:", user\_info)  # Solution starts from here  # Prompt the user to enter a value to delete from the list  value\_to\_delete = input("Please enter a value to delete from the list: ")  # Delete the value from the list if it exists  if value\_to\_delete in user\_info:      user\_info.remove(value\_to\_delete)      print("Value", value\_to\_delete, "has been deleted from the list.")  else:      print("Value", value\_to\_delete, "not found in the list.")  # Display the updated list  print("The updated list of your details is:", user\_info) |

**Activity 4**

1. Present the user with a [list](https://academy.engagelms.com/mod/page/view.php?id=185968) of 7–9 items (such as the [list](https://academy.engagelms.com/mod/page/view.php?id=185968) created in the previous activities).
2. Prompt them to select one value from the [list](https://academy.engagelms.com/mod/page/view.php?id=185968) to update, along with the new value.
3. Change the selected value to the new value and display the updated [list](https://academy.engagelms.com/mod/page/view.php?id=185968) to the user.

|  |
| --- |
| # List to store the user answers  user\_info = []  # Collects the answers from the user  name = input ("what is your name?")  age = input ("How old are you?")  birthday = input ("when is your birthday? Ex. YYYY-MM-DD")  hobby = input ("What is your favorite hobby?")  # Store the answers in the list  user\_info.append(name)  user\_info.append(age)  user\_info.append(birthday)  user\_info.append(hobby)  # Display the results  print("Here are the details you provided:")  print("Name:", user\_info[0])  print("Age:", user\_info[1])  print("Birthday:", user\_info[2])  print("Favorite Hobby:", user\_info[3])  print(user\_info)  # Prompt the user for 2–4 more values to add to the list  value1 = input("Please provide another piece of information about yourself: ")  value2 = input("Please provide one more piece of information about yourself: ")  # Add the new values to the list  user\_info.extend([value1, value2])  # Display the updated list  print("The updated list of your details is:", user\_info)  # Solution starts from here  # Prompt the user to update a value in the list  update\_index = int(input("Please enter the index of the value you want to update (starting from 0): "))  if 0 <= update\_index < len(user\_info):      new\_value = input("Please enter the new value: ")      user\_info[update\_index] = new\_value      print("Value updated successfully.")  else:      print("Invalid index.")  # Display the updated list  print("The updated list of your details is:", user\_info) |

**Tip**

Look up how to identify the index of a specific value in a [list](https://academy.engagelms.com/mod/page/view.php?id=185968).

**Activity 5**

Create four tuples:

1. A tuple with a person's first name and last name.
2. A tuple with the person's current profession and the year they started in it.
3. A tuple with the person's current address.
4. A tuple with the person's previous address.

Combine all tuples into a new, flattened, single tuple that contains all items from the original tuples.

|  |
| --- |
| # Define the tuples  name = ("name", ("Sarah", "Alzahmi"))  profession = ("profession", "Applications Developer", "start\_year", 2019)  current\_address = ("current address", ("United Arab Emirates", "AlDhaid", "UAE"))  previous\_address = ("previous address", ("United Arab Emirates", "Fujairah", "UAE"))  # Combine all tuples into a single flattened tuple  combined\_tuple = (\*name, \*profession, \*current\_address, \*previous\_address)  # Display the combined tuple  print(combined\_tuple) |

**Activity 6**

Create a tuple that contains the following:

* A person's first name and last name as a string.
* A [list](https://academy.engagelms.com/mod/page/view.php?id=185968) containing the titles of the person's favorite movies.

Then perform the following:

1. Display the tuple to the user.
2. Ask the user to provide a new favorite movie to the [list](https://academy.engagelms.com/mod/page/view.php?id=185968).
3. Append the new movie to the [list](https://academy.engagelms.com/mod/page/view.php?id=185968) of favorite movies within the tuple.
4. Display the updated tuple to the user.

While the tuple itself is immutable, the [list](https://academy.engagelms.com/mod/page/view.php?id=185968) within the tuple is mutable, so you can modify its contents.

|  |
| --- |
| # Define the tuple  person\_info = ("Sarah", ["Inception", "The Shawshank Redemption", "The Dark Knight"])  # Display the tuple and prompt the user for a new favorite movie  print("Person's information:")  print("Name:", person\_info[0])  print("Favorite Movies:", person\_info[1])  # Ask the user to provide a new favorite movie  new\_movie = input("Please provide a new favorite movie: ")  # Append the new movie to the list of favorite movies within the tuple  person\_info[1].append(new\_movie)  # Display the updated tuple to the user  print("Updated person's information:")  print("Name:", person\_info[0])  print("Favorite Movies:", person\_info[1]) |