Make a database

# Predict .

Take a look at the code below. Read it carefully and think about what this code will do when executed. What inputs are required? What will be the output based on those inputs?

| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23 | players = []  add\_players = True  while add\_players:  print("Enter a username:")  username = input()  print("Enter a password:")  password = input()  print("Enter a score")  score = input()  player = {"username" : username,  "password" : password,  "score" : score}  players.append(player)  print("Would you like to add another player? Y/N")  answer = input().upper()  if answer == "N":  add\_players = False  print(players) |
| --- | --- |

Write your prediction in the box below:

|  |
| --- |

Run .

**Copy** the code into your development environment and run it.

Was your prediction correct? Did anything unexpected happen? Write down your thoughts below:

|  |
| --- |

Investigate .

| **Questions/activities** | **Your answers** ▿ |
| --- | --- |
| Lines 6, 8, and 10 all prompt for input that will be held in three variables.   * Where else in the program are these three variables being accessed? | Lines 12 to 14. |
| * What data structure is being created at line 12? | A dictionary. |
| * Where is the players list initialised? | Line 1. |
| * Where is new data added to the players list? | Line 16. |
| * **What** new data is being added to the players list on this line? | The dictionary called player. |
| * What condition needs to be True in order for the while loop to terminate? | The user needs to type an N when prompted. |
| Enter **two** new records during execution before typing N when prompted.   * What is the output when line 23 is executed? | The output is the list that contains two items. The items are dictionaries that have the two records that I entered. |

Modify .

| **Modification** | **Hint** |
| --- | --- |
| The programmer has realised that they have left out an attribute from the record. They need to have an attribute for highest\_score.   * Modify the program so that the user can input a highest\_score which is then added to the dictionary (record). | Take a look at lines 9, 10, and 14 to see what new code you need to enter. |
| The program needs to display the dictionary (record) that is located at position 0 of the list.   * At the bottom of the program, write a line of code that will display this. | Remember to test your code by adding two records and seeing if a single dictionary is displayed. |
| The program needs to access the password for the dictionary (record) that is located at position 0.   * At the bottom of the program, write some code that will display the password. | The dictionary in location 0 of the list doesn’t have an identifier.  You can assign it to a variable, so that you can refer to it.  Look back at your work from the ‘Make a record’ activity to see how to access the data pairing for a given key. |
| Test that your program works correctly. | **If you input the following when prompted:**  Fred  House  5  5  N  **The final line of output should be:**  House |
| Modify the program so that the user can choose which record they wish to access by typing in the index. | If the user types a 0, then it should display the entire record at location 0. |
| Modify the program so that the user can choose which attribute they wish to access by typing the name of the attribute. | If the user types highest score, then it should display the highest\_score of their chosen record.  The following is some example input/output that would happen after a record had been entered:  Which record would you like to access?  0  Which attribute would you like to access?  highest score  5  >>> |

Explorer task (make) .

In the **‘Make a record’** **activity** you created a record for your chosen entity. Adapt the program so that it allows for multiple records for that entity to be held in a list. You should use the code from this activity to support you with this.