Computing

Lesson 1: Records and Dictionaries

Programming Part 6: Dictionaries and Datafiles

Rebecca Franks

[1] *Materials from the Teach Computing Curriculum created by the National Centre for Computing Education*

Make a record

Code snippets

Creates a record using the dictionary data structure and prints the entire structure

```
player = {"username": "rockstar",
          "password": "6goatsEating",
          "score": 5328}

print(player)
```

Accesses and prints an attribute within a dictionary data structure using the key

```
print(player["username"])
```

Code snippets

Modifies the data paired with the key

```
player["password"] = "7goatsEating"
```

Displays the data for an attribute within a string

```
print("Username:", player["username"])
```

Task: Create a record

Step 1

Use the dictionary data structure to create a record for an entity of your choosing. Example entities could be a:

- Game
- Book
- Film

Your record should have at least three attributes.

Step 2

Test your dictionary by printing it.

Tip: If there is an error, check the syntax with the code snippet above.

Task: Access and display the data in your dictionary

Step 1

Create several print statements that will print each attribute from your dictionary with a suitable heading. An example for a book record is given below:

```
Title: Harry Potter and the Philosopher's Stone
Author: J K Rowling
ISBN: 0747532745
>>>
```

Step 2

Test your code

Task: Modify the dictionary so that it contains new data

Step 1

Add several lines of code that will change the data paired with all of the attributes in your dictionary. Your program should print the dictionary before it is changed and print it again after.

An example output is below:

```
{'Title': "Harry Potter and the Philosopher's Stone", 'Author': 'J K Rowling', 'ISBN': '0747532745'}
{'Title': "The Official Raspberry Pi Beginner's Guide", 'Author': 'Gareth Halfacree', 'ISBN': '978-1-912047-62-8'}
```

Step 2

Test your code

Explorer Task (Optional)

Modify your program so that it now asks for user input which will then be added to the data paired with each key.

Make a database

Task 1: Predict

Take a look at the code on the next slide. 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?

Remember to write your prediction down.

Task 1: Predict

```
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)
```

Task 2: Run

Open and **run** the file with this code.

Here's a copy of the program (**oaknat.uk/comp-ks4-py-database**).

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

Task 3: Investigate

Investigate the program using the steps below:

Step 1

Lines 6, 8 and 10 all prompt for input that will be held in 3 variables.

- Where else in the program are these three variables being accessed?

Step 2

- What data structure is being created at line 12

Step 3

- Where is the `players` list initialised?

Task 3: Investigate

Investigate the program using the steps below:

Step 4

- Where is new data added to the `players` list?

Step 5

- **What** new data is being added to the `players` list on this line?

Step 6

- **What** condition needs to be `True` in order for the `while` loop to terminate?

Task 3: Investigate

Investigate the program using the steps below:

Step 7

Enter two new records during execution before typing `N` when prompted.

- What is the output when line 23 is executed?

Task 4: Modify

Modification 1	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.

Task 4: Modify

Modification 2	Hint
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.

Task 4: Modify

Modification 3	Hint
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 first task to see how to access the data pairing for a given key.

Task 4: Modify

Modification 4	Hint

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`

Task 4: Modify

Modification 5	Hint
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.

Task 4: Modify

Modification 6	Hint

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 (Optional)

In the first task 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 task to support you with this.

Resume the video now