

Developers Institute

Python Course

Week 1

Day 4

Exercises

Exercise 1 – Birthday Look-up

1. Create a variable called **birthdays**. Its value should be a dictionary.
2. Initialize this variable with birthdays of 5 people of your choice. For each entry in the dictionary, the key should be the person's name, and the value should be their birthday. Use the format "YYYY/MM/DD".
3. Print a welcome message to the user. Then tell him: "You can look up the birthday of the people in the list!"
4. Ask for a person's name ("Which person's birthday do you want to see?") This should be stored in a variable.
5. Get the birthday for the person's name from the **birthdays** dictionary.
6. Print out the birthday with a nicely-formatted message.

Exercise 2 – Birthdays Advanced

1. Extend exercise 1 (use a new python file).
2. Before asking the user to type in a person's name, print out all of the names in the 'database', to make it easier for them to choose.
3. If the person that the user types is not found in the dictionary, print an error message ("Sorry, we don't have birthday information for <person's name>")

Exercise 3 – Add Your Own Birthday

1. Extend exercise 2 (in a new file)
2. Insert new code: before you offer the user to type a person's name to look up, ask the user to add a birthday first:
 1. Ask the user for a person's name – store it in a variable
 2. Ask the user for this person's birthday (in the format "YYYY/MM/DD") - store it in a variable.
 3. Now add this new data into your dictionary.
3. The rest of your code should follow (from exercise 2).
4. Make sure that if the user types any name that's in the dictionary – including the name that she entered herself – the corresponding birthday is found and displayed.

Exercise 4 – Math on a List

1. We want to get a list of 10 numbers from the user:

2. Ask the user for an integer between -100 and 100 – repeat this question 10 times. Each answer from the user should be added onto a list variable that you created earlier.
3. After asking the user 10 times, you should now have a list of integers.
4. Print a line(s) to separate our input section (getting the numbers from the user) from our output section (which we will describe below).
5. We will print some mathematical information about the list of numbers. Each time, print the answer together with a helpful string message that tells the user what you are printing.
6. **NB:** *While we are testing our code, it can get tedious to keep typing in 10 number each time! To save time, you can change '10' to '2' or '3' while you are testing. Just be sure to change it back to 10 once your code works, so that you can test it more thoroughly, and so that the finished code will be correct.*
 1. The list of numbers – printed in a single line
 2. The sum of all the numbers
 3. A list containing the first and the last number only
 4. The numbers without any **duplicates** (**Test** this by typing in some duplicates)
 5. A list of all the numbers greater than 50 (*There are at least 2 ways of doing questions 5-7. Can you do it in the most efficient way?*)
 6. A list of all the numbers smaller than 10
 7. A list of all of the numbers **squared** – eg. for [1, 2, 3] you would print “1 4 9”
 8. (Bonus: The **average** of all the numbers)
 9. (Bonus: The **largest** number)
 10. (Bonus: The **smallest** number)

Exercise 5 – Randoms

1. Extend Exercise 4 (copy it into a new file)
2. Instead of asking the user for 10 integers, generate 10 random integers yourself. Make sure that these random integers lie between -100 and 100.
3. Instead of always generating 10 integers, let the amount of integers also be random! Generate a random positive integer no smaller than 50.
4. Go back and check all of your output! Does your code work correctly for a list of unknown length, or does it only work correctly for a list that has 10 items in it?