

## Looping Through a Dictionary Continued

### Textbook

## Looping through a Dictionary Continued



Loops become particularly useful when you combine them with other code, such as an [if statement](#).

```
1 classmates = {
2     "Billy" : 8,
3     "Vance" : 12,
4     "Alice" : 10,
5     "Eliza" : 15,
6     "Xavier" : 6,
7 }
8
9 for x in classmates:
10     if x == "Eliza":
11         print("This person wants to be anonymous")
12     else:
13         print(x)
```

Try it!

The for loop runs through each student's name and checks to see if the name is Eliza. If it is Eliza, it will print out the string, "This person wants to be anonymous." If it is not Eliza, it will print x. The x in this example would be the keys in the dictionary.

## Using a For Loop to Create a Dictionary

You can also use for loops to create dictionaries. Let's say we were creating a dictionary to say how many people are in all of your classes.

```
1 periods = int(input("How many class periods do you have?"))
2 schedule = {}
3 for i in range(periods):
4     subject = input("What subject?")
5     num_people = input("How many people are in your " + subject + " class?")
6     if subject not in schedule:
7         schedule[subject] = num_people
8 print(schedule)
```

Try it!

For example, this code could print out the following dictionary:

```
{'math': '23', 'science': '12', 'english': '83', 'french': '4'}
```

Notice that we created an empty dictionary named `schedule = {}`.

Using `range()`, we created a for loop that will run as many times as the input named periods.

For each time it loops through the range named periods, it will ask for two inputs: a subject and a number of people in that class.

Since python dictionaries cannot have duplicate keys, we have an if statement to see if that key already exists in the dictionary. If it doesn't, we add it to the dictionary.

After the for loop has run as many times as we determined in the range(), we print the dictionary.

## Checkpoint

### Loop through a Dictionary cont'd

1. Create a dictionary named `ride`.
2. Inside the dictionary, create keys for **5 different people's names**.
3. Pair each key with a number height.
4. Create a **for loop** that loops through the dictionary. For this challenge, use `for x in`
5. Make sure that the loop will check each value.
6. Make the loop first check to see if the student height is greater than or equal to **100**.
7. If it is, print `This person is tall enough to ride.`
8. If the student height is **less than 100**, print `This person is too short to ride.`

## Requirements:

- Create a dictionary named `ride`.
- Create keys for 5 different people's names.
- Loop through each value.
- Create an IF statement inside the loop to check each value.

## Questions (5)

### 1. When an IF statement is inside a FOR loop, how many times will the IF statement run?

MULTIPLE CHOICE

Choose the correct answer:

- A. The IF statement will run each time the program loops.
- B. The IF statement will run for the first and last loop.
- C. The IF statement will run for only loop iterations that meet the condition.
- D. The IF statement cannot run inside a loop.

### 2. True or False: You can create an empty dictionary.

MULTIPLE CHOICE

Choose the correct answer:

- A. True
- B. False

### 3. True or False: Dictionaries can have duplicate keys.

MULTIPLE CHOICE

Choose the correct answer:

- A. True
- B. False

### 4. What will the following code print out?

MULTIPLE CHOICE

```
classmates = { "Billy" : 8, "Vance" : 12, "Alice" : 10, "Eliza" : 15, "Xavier" : 6, } for x in classmates: if x == "Kate": print("This person wants to be anonymous") else: print(x)
```

Choose the correct answer:

- A. Billy Vance Alice Eliza Xavier
- B. 8 12 10 15 6
- C. This person wants to remain anonymous
- D. Kate

### 5. What will the following code print out?

```
classmates = { "Billy" : 8, "Vance" : 12, "Alice" : 10, "Eliza" : 15, "Xavier" : 6, }  
for x in classmates: if x == "Billy": print("This person wants to be anonymous") else: print(x)
```

Choose the correct answer:

- A. This person wants to be anonymous Vance Alice Eliza Xavier
- B. Billy Vance Alice Eliza Xavier
- C. 8 12 10 15 6
- D. This person wants to be anonymous Billy

## Challenges (5)

### 1. Check for a Key

1. Place the following dictionary in your program:

```
dictionary = { 7: "first", 3: "second", 4: "third", 8: "fourth", 9: "fifth" }
```

2. Choose whatever you wish for your values.

3. The program will input a list of integers.

You can get a list from an input like this:

```
my_list = [int(n) for n in input().split()]
```

4. For each of those integers, if it is a key in the dictionary, print " **Yes** ", otherwise print " **No** ".

For example:

Input: **3 9 5 2**

Output: **Yes Yes No No**

Another example:

Input: **2 5 8 4**

Output: **No No Yes Yes**

## 2. Word Frequency

1. Write a program that prints the word that occurs the most frequently in the input. The input is a list of words separated by spaces.
2. Using for loops, create a dictionary that stores the unique words and how many times it has occurred.

**Hint:** create an empty dictionary that you add words and their word count to

For example:

```
animals = { otters: 5, porcupines: 2, badgers: 9 }
```

**Hint:** This is how to add something to a dictionary `dictionaryName[squirrels] = 7`

Print the most frequent word.

If words appear an equal number of times, print both of the most frequent words.

For example:

Input: `apple banana apple apple orange orange`

Output: `apple`

Another example:

Input: `up up down up down up`

Output: `up`

## 3. Power Dictionary

1. Write a program that takes in one input ("How many keys in your dictionary?") and generates a dictionary of that size.
2. The keys and values of the dictionary should be from "0" to "n", with the key being the number and the value being the number multiplied by itself.
3. Print the dictionary.

For example:

Input: `3`

Output: `{0: 0, 1: 1, 2: 4}`

Another example:

Input: `9`

Output: `{0: 0, 1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64}`

## 4. Thesaurus

Now that you've created the dictionary it's time to use it like a thesaurus! You will enter a word and your thesaurus will print out a synonym.

**The final input** is a word that you want to choose to print its synonym.

1. Start by entering an integer that indicates the number of word pairs you want to input.

2. For each word pair, input a word (e.g., "happy") and its synonym (e.g., "delighted"). The program will create thesaurus entries for both the word and its synonym.

3. Once you've completed entering the word pairs, the program will prompt you to enter a word for which you want to find a synonym.

4. Enter the word you want to look up. The program will search the thesaurus and print the synonym associated with the entered word, whether you input the key (e.g., "happy") or its synonym (e.g., "delighted").

For example:

First Input: 2

Second Input: happy

Third Input: delighted

Second Input Repeated: friend

Third Input Repeated: pal

Final Input: delighted

Your code should create a dictionary with the first four words, then print "happy" as the synonym with "delighted".

Output: happy

Another example:

Inputs: 3 , annoy , bother , cold , chilly , allow , permit , bother

Output: annoy

Hint: you need to make sure it will print the synonym if you type in the key or if you type the value. Try this

```
thesaurus[key]=value thesaurus[value]=key
```

Hint: to print out the synonym of a word you type at the end try this

```
thesaurus[input("Which word do you want the synonym for?")]
```

## 5. Student Body President

Your school is having an election! You need to create a program to help track the number of votes for each candidate. The trouble is, all the classes in the school will be voting at the same time! You also don't know the names of the candidates yet nor how many people will be running. So your program needs to account for this.

Each class will be sending you total numbers for their class. They will be giving you the name of each candidate and the number of votes that candidate received in their classroom.

1. Build a program that will help you tally the votes up by using inputs and looping through a dictionary!
2. **The first input** will be the **total number of data points you recieved** at the end of the voting day. This means that each name will be a data point and each group of votes for that name will be a data point. This input will also be the number of times the loop runs.
3. For each time the loop runs, there will be **two inputs**. One input will be the candidate's name as the key. The other input will be the number of votes received as the value. Student names will be repeated because of different classes that voted.
4. But dictionaries aren't allowed to have duplicate keys.
5. Since dictionaries don't allow duplicates, you need to consolidate the votes for the same candidate. Your program should sum all the votes received for each candidate, and print the final totals for each candidate in a dictionary.

*For example:* The first input could be **5**.

6. Then, the next inputs would be the different vote groups collected in 10 different inputs such as the following: `Brown , 10 , Jackson , 8 , Brown , 15 , Jackson , 18 , Jackson , 2`
7. This means that Brown got 10 votes, Jackson got 8 votes, Brown got 15 votes, etc.

From these inputs your code should print `{'Brown': 25, 'Jackson': 28}` .

*For example:*

First Input (the number of times the loop will run): `5`

Inputs for the first time the loop runs. 2 inputs per loop (Name and number of votes): `Brown , 10`

Inputs for the second time the loop runs: `Jackson , 8`

Inputs for the third time the loop runs: `Brown , 15`

Inputs for the fourth time the loop runs: `Jackson , 18`

Inputs for the fifth time the loop runs: `Jackson , 2`

Output: `{'Brown': 25, 'Jackson': 28}`

*Another example:*

Inputs: `4 , Olivia , 2 , Olivia , 2 , Naomi , 4 , Felicity , 5`

Output: `{'Olivia': 4, 'Naomi': 4, 'Felicity': 5}`

## Answer Keys & Solutions

### Checkpoint Solutions

#### Loop through a Dictionary cont'd

```
1 ride = {  
2     "Andrew": 120,  
3     "Francis": 50,  
4     "Zoe": 75,  
5     "Sylvester": 200,  
6     "John": 150,  
7 }  
8  
9  
10 for x in ride.values():  
11     if x >= 100:  
12         print("This person is tall enough to ride")  
13     else:  
14         print("This person is too short to ride")
```

### Questions

#### 1. When an IF statement is inside a FOR loop, how many times will the IF statement run?

MULTIPLE CHOICE

Correct Answer:

- A. The IF statement will run each time the program loops. ✓ Correct
- B. The IF statement will run for the first and last loop. ✗ Incorrect
- C. The IF statement will run for only loop iterations that meet the condition. ✗ Incorrect
- D. The IF statement cannot run inside a loop. ✗ Incorrect

#### Explanation:

Everything inside a for loop will run each time the loop runs.

#### 2. True or False: You can create an empty dictionary.

MULTIPLE CHOICE

Correct Answer:



A. True

✓ Correct

B. False

✗ Incorrect

**Explanation:**

Just like an empty list, you can create an empty dictionary. `mydictionary = { }`

**3. True or False: Dictionaries can have duplicate keys.**

MULTIPLE CHOICE

**Correct Answer:**

A. True

✗ Incorrect

B. False

✓ Correct

**Explanation:**

Dictionaries have only one of each key

**4. What will the following code print out?**

MULTIPLE CHOICE

**Correct Answer:**

A. Billy Vance Alice Eliza Xavier

✓ Correct

B. 8 12 10 15 6

✗ Incorrect

C. This person wants to remain anonymous

✗ Incorrect

D. Kate

✗ Incorrect

**Explanation:**

Kate is not in the list

**5. What will the following code print out?**

MULTIPLE CHOICE

**Correct Answer:**

A. This person wants to be anonymous Vance Alice Eliza Xavier

✓ Correct

B. Billy Vance Alice Eliza Xavier

✗ Incorrect

C. 8 12 10 15 6

✖ Incorrect

D. This person wants to be anonymous Billy

✖ Incorrect

### Explanation:

The key of Billy will be replaced with the print statement

## Challenges

### 1. Check for a Key

Solution:

```
1 dictionary = {
2     7: "first",
3     3: "second",
4     4: "third",
5     8: "fourth",
6     9: "fifth",
7 }
8
9 my_list = [int(n) for n in input().split()]
10
11 for x in my_list:
12
13     if x in dictionary:
14         print("Yes")
15     else:
16         print("No")
```

### 2. Word Frequency

Solution:

```
1 words_count = {}
2
3 words = input("Create a list of words").split()
4 for word in words:
5     if word not in words_count:
6         words_count[word] = 0
7     words_count[word] += 1
8
9 biggest = 0
10
11 for x in words_count.values():
12     if x > biggest:
13         biggest = x
14
15 for n in words_count:
16     if words_count[n] == biggest:
17         print(n)
```

### 3. Power Dictionary

Solution:

```
1 n = int(input("How many keys in dictionary?"))
2
3 dictionary = {}
4 for x in range(n):
5     dictionary[x] = x * x
6
7 print(dictionary)
```

### 4. Thesaurus

Solution:

```
1 n = int(input("Range"))
2
3 dictionary = {}
4 for x in range(n):
5     first = input("First Word")
6     second = input("Synonym")
7     dictionary[first] = second
8     dictionary[second] = first
9
10 print(dictionary[input()])
```

### 5. Student Body President

Solution:

```
1 n = int(input("How many vote sections were collected?"))
2 votes_total = {}
3 for i in range(n):
4     candidate = input("Which candidate?")
5     num_votes = input("How many votes did " + candidate + " get?")
6     if candidate not in votes_total:
7         votes_total[candidate] = 0
8     votes_total[candidate] += int(num_votes)
9 print(votes_total)
```