Using Dictionaries in Python

##### Speaker: Curt Knowles, Instructor

##### [00:00:00.530]

Hello, everyone in this lecture we're going to continue our study in Python and work our way to use a new data structure called dictionaries. Dictionaries are really very similar to lists, which we talked about in our previous lecture. There are a few key differences between dictionaries and lists. First of all, lists consist of collection of single items, whereas dictionaries consist of a collection of key value pairs. Lists are accessed via their position using an index number, but dictionaries are accessed according to their key values.

##### [00:00:45.610]

Lists are in an ordered sequence. They can be sorted. There are sort functions you can use on lists, but dictionary key value pairs are unordered and cannot be sorted. The syntax for creating a dictionary in Python looks similar to a list where you create a variable that will be your dictionary name and you set it equal to some collection of key value pairs. Each key value pair.

##### [00:01:11.580]

The key in the value is separated by a colon. Each key value pair is separated from other key value pairs by commas, and instead of using square brackets as you do in list, you use curly brackets to distinguish the entire collection. Here are some examples of code that creates dictionaries. In the first example, we create a countries dictionary that consists of three key value pairs. In each one of the pairs, the first value is the key that would be the abbreviation of the country, and then the country name is the value.

##### [00:01:54.960]

In the second example, we create a numbers dictionary where our keys are 1,2,3,4, and 5, and our values are those numbers written out as strings, so you can use different data types for your key than what you do for your value. In this case, we use an integer for the key and a string for the value. Here we have an example where we use strings as keys and the values can actually be mixed types. So the movie list consists of a string key with the keys being name, year, and price of the movie.

##### [00:02:34.160]

The name is a string value, the year is an integer value and the price is a float value.

##### [00:02:42.850]

You can create an empty dictionary. To do that, you simply set a variable for your dictionary name equal to the curly bracket opened and closed with nothing in between, and you can print a dictionary using the print statement just like you can do with the list. When you do print this, it will print the key value pairs separated by colons and commas, and it will also print the curly opening and closing curly brackets.

##### [00:03:12.410]

Let's show you some things, some actions you can take on a dictionary. For these examples, we will use our countries dictionary that consist of four entries, key value pairs for Canada, United States, Great Britain, and Mexico. First of all, the way that you access and get a value from the dictionary is by referencing its key. So in this statement, we are going to create a variable called country and into that variable we're going to put the value that has the key MX in our countries dictionary.

##### [00:03:48.660]

So MX key correlates to the value Mexico, so the value Mexico will be placed into country.

##### [00:04:00.050]

In our second example, we are attempting to access a country key that does not exist in our dictionary, and therefore the Python interpreter will give us a Key Error that says key does not exist. If we want to set a value in the key value in the dictionary to a different value, we simply use an assignment statement. On the left hand side. We put the reference to the key that we want to change the value for and to the right side of the assignment statement we place the new value that we want to put into that key.

##### [00:04:43.190]

So GB would now contain United Kingdom instead of Great Britain. Also, if we do an assignment statement where the key that we are assigning does not exist in the country's dictionary, it will place a new key value pair for that key in value into the dictionary. So this last example would place the key "FR" and the value of "France" in the last position, it would append it to the existing countries dictionary.

##### [00:05:19.260]

Just like with lists, we can use the "in" keyword to check for existence of a key in the dictionary.

##### [00:05:29.350]

For example, let's create a variable called "code", and let's set the value of "IE" into code. Then if we say "if code in countries dictionary", it will look for a key that is equal to our code variable, in this case "IE". It will look for

##### [00:05:50.670]

IE In the countries list. If it finds it, it will get us the value that exists for that key., place it in the country variable, and will print that value. If it does not find IE in the countries list, it will execute the else clause and print "There i no country for this code IE".

##### [00:06:14.330]

You can also use the get method of the dictionary object to get the value for the key as well. To use the get method, you specify the name of the dictionary, the dot operator, the word get, and in in parentheses the key that you want to get the value for, and this statement would place the value Mexico into the variable country. If we attempt to get a key that is not in the dictionary, it will not error, but it will just simply place nothing into our country variable.

##### [00:06:56.380]

Alternatively, we can specify a default if the key value is not found. So if we specify a second parameter here with the string "Unknown", that will be the default that will be returned if IE is not found in the dictionary. So this third example would place the string "Unknown" into our country variable.

##### [00:07:22.320]

You can delete items from a dictionary using the del keyword, then the dictionary name and in square brackets this time, the key. So if I say "del countries["MX"]" it will delete the key value pair for the key "MX". It will delete MX and Mexico. If I attempt to delete a country key that is not in the dictionary, it will again give me a key error.

##### [00:07:54.940]

The key doesn't exist. You can you can create a workaround to get around the key error problem by checking first to see if a key exists in the dictionary. To do this, you set your key that you're looking for to delete into a variable. In this case we call it code. Then we use our "if code in countries".

##### [00:08:22.860]

So this is looking for the "IE" key in the countries dictionary. If it is, then get the value and delete that value key pair from the countries dictionary. If that is not found, then we can print there is a no country for this code message. Instead of running into the key error. A second way to delete an item from the dictionary is use

##### [00:08:52.160]

the pop method. So with pop you can simply set a variable equal to countries.pop("US"). It will pop US and United States

##### [00:09:07.850]

key value pair. Delete it from the dictionary and it will place the value United States into the country variable. If you try to pop a key that is not in the list, you will get a key error, but you can get around that by giving it a default value to pop in case it doesn't find that key in the list.

##### [00:09:33.430]

Again, you can prevent the key error from occurring by using the default value for the pop method. You can also clear all items in the list using the clear method. The last example on this slide shows you how to use the clear method by giving the dictionary name, dot, clear and you can't forget the open close parentheses because clear is a function method, so it requires a parameter list even if it is empty.

##### [00:10:06.760]

There are three dictionary methods for getting all keys and values in the list. There are keys, items and values. We're going to look at keys and values in particular.

##### [00:10:21.460]

So if I want to loop through all the keys and values in a list, I can do that using the keys method. I can say "for code in..." - code is a variable I created to hold the next key I'm going to retrieve from the countries dictionary. "for code in countries.keys()". I can print the code and I can print the value that is at that code. Another way to get the same results, because the default iterator in a for loop is keys is, you can just simply say "for code in countries", but if you're going to loop for all the values you have to use the values designation.

##### [00:11:12.990]

Keys is the default looping designation. So if you want to loop just through the values then you have to use the .values method explicitly created. So here we're saying "for name in countries.values()". This will ignore the keys and will loop through all the values in the key value pairs, and it will print only the values.

##### [00:11:41.100]

As you can see with the console output.

##### [00:11:46.010]

There is also a built in constructor for creating dictionaries and lists, so we can have a dictionary and we can convert it to a list, or we can have a list and convert it to a dictionary. We do this using the list and the dict functions. We're going to look specifically at the list. So I'm going to take my countries dictionary and I'm going to convert it to a list. To do this, I create a new list.

##### [00:12:17.680]

I'm going to call it my codes list. I'm going to set codes equal to countries.keys(). That's my keys CA, US, and MX in my countries list. I'm going to convert those into a list and place that list into codes.

##### [00:12:38.090]

Then I can sort - remember, you can sort of list, but you can't sort a dictionary. So now I can call the list sort function that we learned about in the last lecture to sort my new list of keys. And then I can use another for loop to go through each code in the codes list and print out the code from the list and also the value that correlates to that code in my countries dictionary. So what I will get here is the codes list. This statement here that creates the list called codes.

##### [00:13:21.500]

It would create this list here with the value CA, US and MX because what I'm doing is I'm creating a list of the keys in the country's table CA, US, and MX. So it creates this codes list. And then when I get to the for loop, I actually the sort statement actually sorts this. So it turns this into CA, MAx, and US would be last.

##### [00:13:53.540]

Now I go through my codes list for each code in codes, I'm going to print the code. So the first one I get to is CA - I'm going to print the code CA, and then I'm going to print the value in the countries dictionary that has the key CA. So I print CA from my list, and then I go to my countries dictionary and see which value has the CA key, and that's Canada and I print that. I do this each time through each code in my codes list.

##### [00:14:30.660]

Remember, it's been sorted. So now as CA, then MX, and then US.

##### [00:14:38.890]

Okay. And one final thing that I wanted to show you was dictionaries and lists can be used to you can start seeing how they can be used when you get into things like phone books or student grade scores. So what you can do is you can create dictionaries inside of dictionaries, or you can create lists inside of lists. So here's a classic example of using dictionaries in Python, and that is to create an address list. So here you actually have one dictionary called contacts.

##### [00:15:17.210]

Okay. And you set contacts equal to key value pairs. And the key is the name of the contact - Joel, Anne, and Ben. The value for each key is actually another dictionary. This entire dictionary here and that dictionary consists of 5 key value pairs - address, city, state, postal code, and phone.

##### [00:15:51.980]

So this is one dictionary inside of another dictionary. So if I wanted to in this case, I wanted to say get me Anne's phone number. The way I would do that in Python is I would go to my contacts dictionary. And first, I would get Anne's record or key value pair. And then I would look into Anne's value dictionary, and I would get her phone key value pair value.

##### [00:16:28.900]

So I would say contact["Anne"]["phone"], and that would get the phone number that is at Anne's record in my dictionary.

##### [00:16:41.750]

And that concludes this brief introduction to dictionaries in Python. I'm going to be presenting another video in this lesson module that will also show you - I will walk through a demo of a problem using dictionaries, and I will create that in Python code for you so you can see how we create and use dictionaries actually in Python code. Thank you very much. Bye bye.