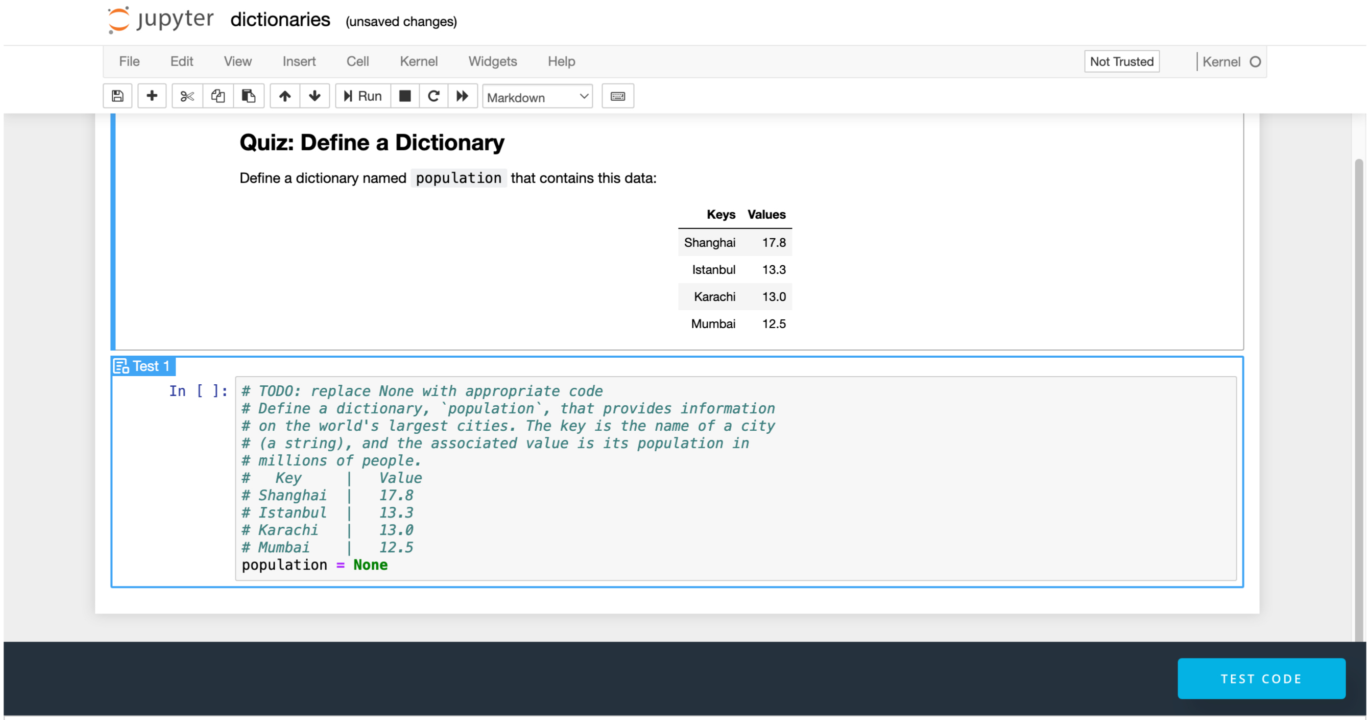
Quiz: Dictionaries and Identity Operators


jupyter dictionaries (unsaved changes)
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Kernel
Run
Markdown
Test 1
TEST CODE

Quiz: Define a Dictionary

Define a dictionary named `population` that contains this data:

Keys	Values
Shanghai	17.8
Istanbul	13.3
Karachi	13.0
Mumbai	12.5

```
In [ ]: # TODO: replace None with appropriate code
        # Define a dictionary, `population`, that provides information
        # on the world's largest cities. The key is the name of a city
        # (a string), and the associated value is its population in
        # millions of people.
        #   Key     |   Value
        # Shanghai  |   17.8
        # Istanbul  |   13.3
        # Karachi   |   13.0
        # Mumbai    |   12.5
        population = None
```

ⓘ Menu ⤢ Expand

Quiz Question

Which of these could be used as the key for a dictionary? (Choose all that apply.) Hint: Dictionary keys must be immutable.

□ `str`

□ `list`

□ `int`

□ `float`

Submit

Quiz Question

What happens if we look up a value that isn't in the dictionary? Create a test dictionary and use the square brackets to look up a value that you haven't defined. What happens?

○ The lookup returns `None`

○ The key is added to the dictionary with a default value of None

○ A `KeyError` occurs

○ Python searches the Internet for an appropriate value

Submit

`get` with a Default Value

Dictionaries have a related method that's also useful, `get()`. `get()` looks up values in a dictionary, but unlike looking up values with square brackets, `get()` returns `None` (or a default value of your choice) if the key isn't found. If you expect lookups to sometimes fail, `get()` might be a better tool than normal square bracket lookups.

```
>>> elements.get('dilithium')
None
>>> elements['dilithium']
KeyError: 'dilithium'
>>> elements.get('kryptonite', 'There\'s no such element!')
"There's no such element!"
```

In the last example we specified a default value (the string 'There's no such element!') to be returned instead of `None` when the key is not found.

Checking for Equality vs. Identity: `==` vs. `is`

Quiz Question

What will the output of the following code be? (Treat the commas in the multiple choice answers as newlines.)

```
a = [1, 2, 3]
b = a
c = [1, 2, 3]

print(a == b)
print(a is b)
print(a == c)
print(a is c)
```

If you want to run test code, use the coding space from earlier in the lesson.

○ True, True, True, True

○ True, False, True, False

○ True, True, True, False

○ True, True, False, False

Submit

← Previous   Next →   Give Page Feedback