Python Dictionary Comprehension

Summary: in this tutorial, you'll learn about Python dictionary comprehension to transform or filter items in a dictionary.

Introduction to Python dictionary comprehension

A dictionary comprehension allows you to run a `for` loop (https://www.pythontutorial.net/python-basics/python-for-range/) on a dictionary (https://www.pythontutorial.net/python-basics/python-dictionary/) and do something on each item like transforming or filtering, and returns a new dictionary.

Unlike a `for` loop, a dictionary comprehension offers a more expressive and concise syntax when you use it correctly.

Here is the general syntax for a dictionary comprehension:

```
{key:value for (key,value) in dict.items() if condition}
```

This dictionary comprehension expression returns a new dictionary whose item specified by the expression `key: value`

Python dictionary comprehension examples

We'll take a look at how to use the dictionary comprehension to transform and filter items in a dictionary.

1) Using Python dictionary comprehension to transform a dictionary

Suppose that you have the following dictionary whose items are stock symbol and price:

```
stocks = {
    'AAPL': 121,
    'AMZN': 3380,
    'MSFT': 219,
    'BIIB': 280,
    'QDEL': 266,
    'LVGO': 144
}
```

To increase the price of each stock by 2%, you may come up with a for loop like this:

```
stocks = {
    'AAPL': 121,
    'AMZN': 3380,
    'MSFT': 219,
    'BIIB': 280,
    'QDEL': 266,
    'LVGO': 144
}

new_stocks = {}
for symbol, price in stocks.items():
    new_stocks[symbol] = price*1.02

print(new_stocks)
```

Output:

```
{'AAPL': 123.42, 'AMZN': 3447.6, 'MSFT': 223.38, 'BIIB': 285.6, 'QDEL': 271.32, '
```

How it works.

- First, loop over the items of the `stocks` dictionary
- Second, increase the price by 2% and add the item to the new dictionary (`new_stocks`).

The following example shows how to use the dictionary comprehension to achieve the same result:

```
stocks = {
    'AAPL': 121,
    'AMZN': 3380,
    'MSFT': 219,
    'BIIB': 280,
    'QDEL': 266,
    'LVGO': 144
}

new_stocks = {symbol: price * 1.02 for (symbol, price) in stocks.items()}

print(new_stocks)
```

This dictionary comprehension is equivalent to the for loop counterpart:

for loop

```
new_stocks = {}
for symbol, price in stocks.items():
    new_stocks[symbol] = price*1.02
```

dictionary comprehension

```
new_stocks = {symbol: price * 1.02 for (symbol, price) in stocks.items()}
```

2) Using Python dictionary comprehension to filter a dictionary

To select stocks whose prices are greater than 200, you may use the following `for` loop:

```
stocks = {
    'AAPL': 121,
    'AMZN': 3380,
    'MSFT': 219,
    'BIIB': 280,
    'QDEL': 266,
    'LVGO': 144
}

selected_stocks = {}
for symbol, price in stocks.items():
    if price > 200:
        selected_stocks[symbol] = price

print(selected_stocks)
```

How it works.

- First, iterate over the item of the stocks dictionary
- Then, add the item to the selected_stocks dictionary if the price is greater than 200

The following example uses the dictionary comprehension with an if clause to get the same result:

```
stocks = {
    'AAPL': 121,
    'AMZN': 3380,
```

```
    'MSFT': 219,
    'BIIB': 280,
    'QDEL': 266,
    'LVGO': 144
}

selected_stocks = {s: p for (s, p) in stocks.items() if p > 200}

print(selected_stocks)
```

And you can compare between the for loop and dictionary comprehension:

for loop

```
selected_stocks = {}
for symbol, price in stocks.items():
    if price > 200:
        selected_stocks[symbol] = price
```

dictionary comprehension

```
selected_stocks = {s: p for (s, p) in stocks.items() if p > 200}
```

Summary

- A dictionary comprehension iterates over items of a dictionary and allows you to create a new dictionary by transforming or filtering each item.