How to Filter List Elements in Python

Summary: in this tutorial, you'll learn how to filter list elements by using the built-in Python `filter()` function.

Introduction to Python filter() function

Sometimes, you need to iterate over elements of a list (https://www.pythontutorial.net/python-basics/python-for-loop-list/) and select some of them based on specified criteria.

Suppose that you have the following list of `scores`:

```
scores = [70, 60, 80, 90, 50]
```

To get all elements from the `scores` list where each element is greater than or equal to 70, you use the following code:

```
scores = [70, 60, 80, 90, 50]

filtered = []
```

```python
for score in scores:
    if score >= 70:
        filtered.append(score)

print(filtered)
```

How it works.

- First, define an empty list (`filtered`) that will hold the elements from the `scores` list.
- Second, iterate over the elements of the `scores` list. If the element is greater than or equal to 70, add it to the `filtered` list.
- Third, show the `filtered` list to the screen.

Python has a built-in function called `filter()` that allows you to filter a list (or a tuple) in a more beautiful way.

The following shows the syntax of the `filter()` function:

```python
filter(fn, list)
```

The `filter()` function iterates over the elements of the `list` and applies the `fn()` function to each element. It returns an iterator for the elements where the `fn()` returns `True`.

In fact, you can pass any iterable to the second argument of the `filter()` function, not just a list.

The following shows how to use the `filter()` function to return a list of `scores` where each score is greater than or equal to 70:

```python
scores = [70, 60, 80, 90, 50]
filtered = filter(lambda score: score >= 70, scores)

print(list(filtered))
```

Output:

```
[70, 80, 90]
```

Since the `filter()` function returns an iterator, you can use a `for` loop to iterate over it. Or you can use the `list()` function to convert the iterator to a list.

Using the Python filter() function to filter a list of tuples example

Suppose you have the following list of tuples:

```
countries = [
    ['China', 1394015977],
    ['United States', 329877505],
    ['India', 1326093247],
    ['Indonesia', 267026366],
    ['Bangladesh', 162650853],
    ['Pakistan', 233500636],
    ['Nigeria', 214028302],
    ['Brazil', 21171597],
    ['Russia', 141722205],
    ['Mexico', 128649565]
]
```

Each element in a list is a tuple that contains the country name and population.

To get all the countries whose populations are greater than 300 million, you can use the `filter()` function as follows:

```
countries = [
    ['China', 1394015977],
    ['United States', 329877505],
    ['India', 1326093247],
    ['Indonesia', 267026366],
    ['Bangladesh', 162650853],
    ['Pakistan', 233500636],
    ['Nigeria', 214028302],
    ['Brazil', 21171597],
    ['Russia', 141722205],
    ['Mexico', 128649565]
]

populated = filter(lambda c: c[1] > 300000000, countries)

print(list(populated))
```

Output:

```
[['China', 1394015977], ['India', 1326093247], ['United States', 329877505]]
```

Summary

- Use the Python `filter()` function to filter a list (or a tuple).