How to Convert Pandas DataFrame into a List

July 7, 2020

At times, you may need to convert *Pandas* DataFrame into a list in Python.

But how would you do that?

To accomplish this task, you can use *tolist* as follows:

```
df.values.tolist()
```

In this short guide, I'll show you an example of using tolist to convert Pandas DataFrame into a list.

Example of using tolist to Convert Pandas DataFrame into a List

Let's say that you have the following data about products and prices:

Product	Price
Tablet	250
iPhone	800
Laptop	1200
Monitor	300

You then decided to capture that data in Python using Pandas DataFrame.

At a certain point, you realize that you'd like to convert that Pandas DataFrame into a list.

To accomplish this goal, you may use the following Python code, which will allow you to convert the DataFrame into a list, where:

- The top part of the code, contains the syntax to create the DataFrame with our data about products and prices
- The bottom part of the code converts the DataFrame into a list using: *df.values.tolist()*

Here is the full Python code:

```
import pandas as pd

products = {'Product': ['Tablet','iPhone','Laptop','Monitor'],
            'Price': [250,800,1200,300]
            }

df = pd.DataFrame(products, columns= ['Product', 'Price'])

products_list = df.values.tolist()
print (products_list)
```

And once you run the code, you'll get the following multi-dimensional list (i.e., list of lists):

```
[['Tablet', 250], ['iPhone', 800], ['Laptop', 1200], ['Monitor', 300]]
```

But what about the column names?

If you want to add the column names into your list, you'll need to modify the code as follows:

```
import pandas as pd

products = {'Product': ['Tablet','iPhone','Laptop','Monitor'],
            'Price': [250,800,1200,300]
            }

df = pd.DataFrame(products, columns= ['Product', 'Price'])

products_list = [df.columns.values.tolist()] + df.values.tolist()
print (products_list)
```

This is what you'll see once you run the code:

```
[['Product', 'Price'], ['Tablet', 250], ['iPhone', 800], ['Laptop', 1200], ['Monitor', 300]]
```

Honestly, between you and me, this list looks ugly.

Let's do some formatting to display the list in a tabular form:

```
import pandas as pd

products = {'Product': ['Tablet','iPhone','Laptop','Monitor'],
            'Price': [250,800,1200,300]
            }

df = pd.DataFrame(products, columns= ['Product', 'Price'])

products_list = [df.columns.values.tolist()] + df.values.tolist()
f = '{:<8}|{:<15}' # formatting

for i in products_list:
    print(f.format(*i))
```

And the result:

```
Product |Price
Tablet  |250
iPhone  |800
Laptop  |1200
Monitor |300
```

Much better!

So once you have your list nicely formatted, you may perform some additional actions, such as appending values to the list.

For example, let's say that you have another product (e.g., a Printer for a price of $150) and you want to append it to the list.

Here is the Python code that you may use:

```
import pandas as pd

products = {'Product': ['Tablet','iPhone','Laptop','Monitor'],
            'Price': [250,800,1200,300]
            }

df = pd.DataFrame(products, columns= ['Product', 'Price'])
```

```
products_list = [df.columns.values.tolist()] + df.values.tolist()
products_list.append(['Printer',150])
f = '{:<8}|{:<15}' # format

for i in products_list:
    print(f.format(*i))
```

The Printer, with a price of $150, would now get appended to the list:

```
Product |Price
Tablet  |250
iPhone  |800
Laptop  |1200
Monitor |300
Printer |150
```

Convert an Individual Column in the DataFrame into a List

Let's say that you'd like to convert the 'Product' column into a list.

You can then use the following template in order to convert an individual column in the DataFrame into a list:

```
df['column name'].values.tolist()
```

Here is the complete Python code to convert the 'Product' column into a list:

```
import pandas as pd

products = {'Product': ['Tablet','iPhone','Laptop','Monitor'],
            'Price': [250,800,1200,300]
            }

df = pd.DataFrame(products, columns= ['Product', 'Price'])

product = df['Product'].values.tolist()
print (product)
```

Run the code, and you'll get the following list:

```
['Tablet', 'iPhone', 'Laptop', 'Monitor']
```

What if you want to append an additional item (e.g., Printer) into the 'Product' list?

In that case, simply add the following syntax:

```
product.append('Printer')
```

So the full Python code would look like this:

```
import pandas as pd

products = {'Product': ['Tablet','iPhone','Laptop','Monitor'],
            'Price': [250,800,1200,300]
            }

df = pd.DataFrame(products, columns= ['Product', 'Price'])

product = df['Product'].values.tolist()
product.append('Printer')

print (product)
```

You'll now see the 'Printer' at the end of the list:

```
['Tablet', 'iPhone', 'Laptop', 'Monitor', 'Printer']
```

An Opposite Scenario

Sometimes, you may face an opposite situation, where you'll need to convert a list to a DataFrame. If that's the case, you may want to check the following source that explains how to convert a list to a DataFrame in Python.

Python
< How to Export DataFrame to CSV in R
> How to Create While Loop in Python (with 4 Examples)