

# pandas\_combining\_data\_frames\_workbook\_answers

September 22, 2020

## 1 PANDAS COMBINING DATA FRAMES WORKBOOK

Remember, there are different ways to write code to get the same answer, so your answer can be correct and different to the answer example!

If you feel stuck and want some in person help, then have a look at the events page to join in a workshop <https://swamphen.co.uk/events>.

```
In [1]: # import numpy and pandas
```

```
import pandas as pd
import numpy as np
```

```
In [2]: # make a 3 by 3 test array
```

```
test = pd.DataFrame({'A': [1,2,3], 'B': [4,5,6], 'C': [7,8,9]})
```

```
In [5]: # append a copy of this array to the bottom of this array
```

```
# i.e. extend the columns
# give it a new name
test_new = test.append(test)
print(test_new)
```

```
   A  B  C
0  1  4  7
1  2  5  8
2  3  6  9
0  1  4  7
1  2  5  8
2  3  6  9
```

```
In [6]: # if you haven't done so already sort out the indexes
```

```
test_new = test.append(test, ignore_index = True)
print(test_new)
```

```
   A  B  C
0  1  4  7
1  2  5  8
2  3  6  9
```

```
3 1 4 7
4 2 5 8
5 3 6 9
```

```
In [7]: # add this new data frame to itself on the right hand side
        # i.e. extend the rows
        # give it a new name
        test_double = pd.concat([test_new, test_new], axis = 1)
        print(test_double)
```

```
   A  B  C  A  B  C
0  1  4  7  1  4  7
1  2  5  8  2  5  8
2  3  6  9  3  6  9
3  1  4  7  1  4  7
4  2  5  8  2  5  8
5  3  6  9  3  6  9
```

```
In [14]: # change the column titles to ABCDEF
         name = list('ABCDEF')
         print(name)
         test_double.columns = name
         print(test_double)
```

```
['A', 'B', 'C', 'D', 'E', 'F']
   A  B  C  D  E  F
0  1  4  7  1  4  7
1  2  5  8  2  5  8
2  3  6  9  3  6  9
3  1  4  7  1  4  7
4  2  5  8  2  5  8
5  3  6  9  3  6  9
```

```
In [16]: # add the first data frame you made to the bottom of this latest
         # data frame, which is 4 copies of the original
         # give it a new name
         test_three = test_double.append(test, ignore_index = True, sort = False)
         print(test_three)
```

```
   A  B  C   D   E   F
0  1  4  7  1.0  4.0  7.0
1  2  5  8  2.0  5.0  8.0
2  3  6  9  3.0  6.0  9.0
3  1  4  7  1.0  4.0  7.0
4  2  5  8  2.0  5.0  8.0
5  3  6  9  3.0  6.0  9.0
```

```
6 1 4 7 NaN NaN NaN
7 2 5 8 NaN NaN NaN
8 3 6 9 NaN NaN NaN
```

```
In [18]: # create a new data frame to append to the end of this latest data frame
# where the values will only be in columns D to F
# give it as many rows as you want to
df = pd.DataFrame({'D':[10], 'E':[11], 'F':[12]})
test_four = test_three.append(df, ignore_index = True, sort = False)
print(test_four)
```

	A	B	C	D	E	F
0	1.0	4.0	7.0	1.0	4.0	7.0
1	2.0	5.0	8.0	2.0	5.0	8.0
2	3.0	6.0	9.0	3.0	6.0	9.0
3	1.0	4.0	7.0	1.0	4.0	7.0
4	2.0	5.0	8.0	2.0	5.0	8.0
5	3.0	6.0	9.0	3.0	6.0	9.0
6	1.0	4.0	7.0	NaN	NaN	NaN
7	2.0	5.0	8.0	NaN	NaN	NaN
8	3.0	6.0	9.0	NaN	NaN	NaN
9	NaN	NaN	NaN	10.0	11.0	12.0