

Applied Data Analytics

Pandas basics

Missing data

Hans-Martin von Gaudecker and Aapo Stenhammar

No income data for Bob and Derek

Name	Income
Alice	3000.0
Bob	
Charlie	5000.0
Derek	

Use pd.NA, take missings into account

```
[1] income = pd.Series(  
    data=[3000.0, pd.NA, 5000.0, pd.NA],  
    index=["Alice", "Bob", "Charlie", "Derek"],  
    name="Income",  
)  
income
```

```
[1] Alice      3000.0  
    Bob        <NA>  
    Charlie   5000.0  
    Derek        <NA>  
    Name: Income, dtype: object
```

```
[2] income.value_counts()  
[2] 3000.0      1  
    5000.0      1  
    Name: count, dtype: int64
```

```
[3] income.value_counts(  
    dropna=False  
)  
[3] <NA>      2  
    3000.0    1  
    5000.0    1  
    Name: count, dtype: int64
```

Check for (non-)missing values

```
[4] income.isna()
```

```
[4] Alice      False  
    Bob       True  
    Charlie  False  
    Derek    True  
    Name: Income, dtype: bool
```

```
[5] income.notna()
```

```
[5] Alice      True  
    Bob       False  
    Charlie  True  
    Derek    False  
    Name: Income, dtype: bool
```

Explicitly exclude missing data

```
[6] income.loc[income.notna()]
```

```
[6] Alice      3000.0  
    Charlie    5000.0  
    Name: Income, dtype: object
```

Fill one value with missing data

```
[7] income.loc["Bob"] = 1000.0  
income
```

```
[7] Alice      3000.0  
     Bob       1000.0  
     Charlie   5000.0  
     Derek      <NA>  
     Name: Income, dtype: object
```

This changes the existing Series.

Fill all values with missing data

```
[8] income_imputed = income.fillna(2000.0)
     income_imputed
```

```
[8] Alice      3000.0
     Bob       1000.0
     Charlie  5000.0
     Derek    2000.0
     Name: Income, dtype: object
```

```
[9] income
```

```
[9] Alice      3000.0
     Bob       1000.0
     Charlie  5000.0
     Derek    <NA>
     Name: Income, dtype: object
```

The existing Series is unchanged.