Cleaning Data

- You can update or replace instances of text in a column
  - You’ll need to make sure the relevant columns are strings
  - If necessary, cast to a string using the *astype()* method

- To replace text, use the *str.replace* method

- For example, to replace the $ character in the *description* column of a DataFrame *df*

```
df[”description"] = df[”description"].str.replace(‘$','')
```

Dealing with Missing Values

- Many datasets have missing values in particular columns

- Pandas provides multiple ways of dealing with this. The easiest way is to just drop the rows with the missing values

```
df.dropna(inplace = True)
#inplace means the changes are made in the DataFrame itself
```

- To drop rows with missing values in particular columns, use the *subset* parameter

```
df.dropna(inplace = True, subset = [‘column 1’, ‘column 2’])
#subset is a list of columns to consider when looking for missing values
```

- Another way is to fill-in the missing values using *fillna()*

```
df.fillna(0)
#the 0 means all NaN (Not a Number) elements will be replaced with 0s
```