# Data Analysis with Python

## Cheat Sheet: Importing Data Sets

| Package/Method | Description | Code Example |
|---|---|---|
| Read CSV data set | Read the CSV file containing a data set to a pandas data frame | `df = pd.read_csv(<CSV_path>, header = None)`<br>`# load without header`<br>`df = pd.read_csv(<CSV_path>, header = 0)`<br>`# load using first row as header`<br>Copied!<br>Note: The labs in this course run in JupyterLite environment. In JupyterLite environment, you'll need to download the required file to the local environment and then use the local path to the file as the CSV_path. However, in case you are using JupyterLabs, or any other Python compiler on your local machine, you can use the URL of the required file directly as the CSV_path. |
| Print first few entries | Print the first few entries (default 5) of the pandas data frame | `df.head(n) #n=number of entries; default 5`<br>Copied! |
| Print last few entries | Print the last few entries (default 5) of the pandas data frame | `df.tail(n) #n=number of entries; default 5`<br>Copied! |
| Assign header names | Assign appropriate header names to the data frame | `df.columns = headers`<br>Copied! |
| Replace "?" with NaN | Replace the entries "?" with NaN entry from Numpy library | `df = df.replace("?", np.nan)`<br>Copied! |
| Retrieve data types | Retrieve the data types of the data frame columns | `df.dtypes`<br>Copied! |
| Retrieve statistical description | Retrieve the statistical description of the data set. Defaults use is for only numerical data types. Use include="all" to create summary for all variables | `df.describe() #default use df.describe(include="all")`<br>Copied! |
| Retrieve data set summary | Retrieve the summary of the data set being used, from the data frame | `df.info()`<br>Copied! |
| Save data frame to CSV | Save the processed data frame to a CSV file with a specified path | `df.to_csv(<output CSV path>)`<br>Copied! |

Skills Network