

Data Analysis with Python

Cheat Sheet: Importing Data Sets

Package/Method	Description
Read CSV data set	Read the CSV file containing a data set to a pandas data frame
Print first few entries	Print the first few entries (default 5) of the pandas data frame
Print last few entries	Print the last few entries (default 5) of the pandas data frame
Assign header names	Assign appropriate header names to the data frame
Replace "?" with NaN	Replace the entries "?" with NaN entry from Numpy library
Retrieve data types	Retrieve the data types of the data frame columns
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
Retrieve data set summary	Retrieve the summary of the data set being used, from the data frame
Save data frame to CSV	Save the processed data frame to a CSV file with a specified path

Code Example

```
1. 1
1. df = pd.read_csv(csv_path, header = None)
# Read without header
df = pd.read_csv(csv_path, header = 0)
# Read using first row as header
```

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.

```
1. 1
1. df.head(n) #number of entries; default 5
```

Copy

```
1. 1
1. df.tail(n) #number of entries; default 5
```

Copy

```
1. 1
1. df.columns = headers
```

Copy

```
1. 1
1. df = df.replace("?", np.nan)
```

Copy

```
1. 1
1. df.dtypes
```

Copy

```
1. 1
1. df.describe() #default use df.describe(include="all")
```

Copy

```
1. 1
1. df.info()
```

Copy

```
1. 1
1. df.to_csv(output_csv_path)
```

Copy

