COMPUTING SUBJECT:	Machine Learning
TYPE:	WORK ASSIGNMENT
IDENTIFICATION:	Missing data & linear regression
COPYRIGHT:	*Jens Peter Andersen & Michael Claudius*
DEGREE OF DIFFICULTY:	Easy
TIME CONSUMPTION:	1 hour
EXTENT:	< 60 lines
OBJECTIVE:	Using a Dataframe with missing data
	Using Scikit-Learn simple imputer

COMMANDS:

The Mission
Establishing a dataframe, which typically is the starting point for machine learning. Using Scikit-Learn's simple imputer to 'purify' data.

The problem
To do find the best regression line for at training set of click data with missing data.

Useful links
https://www.w3schools.com/Python/default.asp
https://docs.python.org/3/library/random.html
https://www.tutorialspoint.com/python_data_structure/python_2darray.htm

Step 1: Establish a Dataframe

Start Jupyter Notebook and make a new notebook: LinearRegMissingData

Import needed libraries:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
```

Establish training set as a dataframe:

```
clickData = {'CostPerClick': [2.3, 2.1, 2.5, 4.5, 5.9, 4.1, 8.9],
             'TotalClicksPerDay': [89.0,63.0,71.0,np.NaN,80.0,89.0,150.0]}
trainingSet = pd.DataFrame(clickData)
trainingSet
```

Step 2: Keep index and columns

Keep index:

```
keptIndex=trainingSet.index
keptIndex
```

Keep columns:

```
keptColumns=trainingSet.columns
keptColumns
```

Step 3: Perform data cleaning

Create simple imputer in order to clean data:

```
#Missing import of SimpleImputer, find out your self

imputer = SimpleImputer(strategy="median")
imputer.fit(trainingSet)
cleanedData=imputer.transform(trainingSet)
cleanedData
```

Note what happened!

Establish cleaned dataset as a Dataframe:

```
trainingSetCleaned=pd.DataFrame(cleanedData,columns=keptColumns,
index=keptIndex)
trainingSetCleaned
```

Step 4: Calculate regression line and plot result

ReUse previous code from *Simple dataframe & linear regression* exercise

Congratulations.