DATA PREPROCESCING

Cleaning the Date

	Reason_1	Reason_2	Reason_3	Reason_4	Date	Transportation Expense	Distance to Work	Age
0	0	0	0	1	07/07/2015	289	36	33
1	0	0	0	0	14/07/2015	118	13	50
2	0	0	0	1	15/07/2015	179	51	38
3	1	0	0	0	16/07/2015	279	5	39
4	0	0	0	1	23/07/2015	289	36	33
5	0	0	0	1	10/07/2015	179	51	38
6	0	0	0	1	17/07/2015	361	52	28
7	0	0	0	1	24/07/2015	260	50	36
8	0	0	1	0	06/07/2015	155	12	34

	Reason_1	Reason_2	Reason_3	Reason_4	Month Value	Day of the Week	Transportation Expense	Distance to Work	Age
0	0	0	0	1	7	1	289	36	33
1	0	0	0	0	7	1	118	13	50
2	0	0	0	1	7	2	179	51	38
3	1	0	0	0	7	3	279	5	39
4	0	0	0	1	7	3	289	36	33
5	0	0	0	1	10	2	179	51	38
6	0	0	0	1	7	4	361	52	28
7	0	0	0	1	7	4	260	50	36
8	0	0	1	0	6	6	155	12	34

1. convert the 'Date' column into datetime

```
df['Date'] = pd.to_datetime(df['Date'], format='%d/%m/%Y')
```

2. create a list with month values retrieved from the 'Date' column

```
list_months = []
for i in range(df.shape[0]):
    list_months.append(df['Date'][i].month)
```

3. insert the values in a new column in df, called 'Month Value'

```
df['Month Value'] = list_months
```

4. create a new feature called 'Day of the Week'

```
df['Day of the Week'] = df['Date'].apply(lambda x: x.weekday())
```

5. drop the 'Date' column from df

```
df = df.drop(['Date'], axis = 1)
```

6. re-order the columns in df

```
column_names_upd = ['Reason_1', 'Reason_2', 'Reason_3', 'Reason_4', 'Month Value', 'Day of the Week',
                    'Transportation Expense', 'Distance to Work', 'Age',
                    'Daily Work Load Average', 'Body Mass Index', 'Education', 'Children',
                    'Pet', 'Absenteeism Time in Hours']
df = df[column_names_upd]
```

DATA PREPROCESCING

Cleaning the ‘Education’

Education	Children	Pet	Absenteeism Time in Hours
1	2	1	4
1	1	0	0
1	0	0	2
1	2	0	4
1	2	1	2
1	0	0	2
1	1	4	8
1	4	0	4
1	2	0	40
3	1	1	8

Education	Children	Pet	Absenteeism Time in Hours
0	2	1	4
0	1	0	0
0	0	0	2
0	2	0	4
0	2	1	2
0	0	0	2
0	1	4	8
0	4	0	4
0	2	0	40
1	1	1	8

1	High school	538
2	Graduate	73
3	Postgraduate	40
4	A master or a doctor	4

1. map 'Education' variables; the result is a dummy

```
df['Education'] = df['Education'].map({1:0, 2:1, 3:1, 4:1})
```

0	High school	538
1	Not High School	117