In [5]:

```
import pandas as pd
import numpy as np
df = pd.DataFrame({'From_To': ['LoNDon_paris', 'MAdrid_miLAN',
'londON_StockhOlm',

'Budapest_PaRis', 'Brussels_londOn'],
'FlightNumber': [10045, np.nan, 10065, np.nan, 10085],
'RecentDelays': [[23, 47], [], [24, 43, 87], [13], [67, 32]],
'Airline': ['KLM(!)', '<Air France> (12)', '(British Airways. )',
'12. Air France', '"Swiss Air"']})
df
```

Out[5]:

	From_To	FlightNumber	RecentDelays	Airline
0	LoNDon_paris	10045.0	[23, 47]	KLM(!)
1	MAdrid_miLAN	NaN	[]	<Air France> (12)
2	londON_StockhOlm	10065.0	[24, 43, 87]	(British Airways.)
3	Budapest_PaRis	NaN	[13]	12. Air France
4	Brussels_londOn	10085.0	[67, 32]	"Swiss Air"

```
##Q1.Some values in the the FlightNumber column are missing. These numbers are
meant to increase by 10 with each row so 10055 and 10075 need to be put in
place. Fill in these missing numbers and make the column an integer column
(instead of a float column)
```

In [10]:

```
df['FlightNumber']=df['FlightNumber'].interpolate().astype(int)
df
```

Out[10]:

	From_To	FlightNumber	RecentDelays	Airline
0	LoNDon_paris	10045	[23, 47]	KLM(!)
1	MAdrid_miLAN	10055	[]	<Air France> (12)
2	londON_StockhOlm	10065	[24, 43, 87]	(British Airways.)
3	Budapest_PaRis	10075	[13]	12. Air France
4	Brussels_londOn	10085	[67, 32]	"Swiss Air"

##Q2.The From_To column would be better as two separate columns! Split each string on the underscore delimiter _ to give a new temporary DataFrame with the correct values. Assign the correct column names to this temporary DataFrame.

In [11]:

```
#temperory dataframe
df1=df
```

In [13]:

```
df1[['Depart','Arrival']]=df1['From_To'].str.split('_',expand=True)
df1
```

Out[13]:

	From_To	FlightNumber	RecentDelays	Airline	Depart	Arrival
0	LoNDon_paris	10045	[23, 47]	KLM(!)	LoNDon	paris
1	MAdrid_miLAN	10055	[]	<Air France> (12)	MAdrid	miLAN
2	londON_StockhOlm	10065	[24, 43, 87]	(British Airways.)	londON	StockhOlm
3	Budapest_PaRis	10075	[13]	12. Air France	Budapest	PaRis
4	Brussels_londOn	10085	[67, 32]	"Swiss Air"	Brussels	londOn

##Q3.Notice how the capitalisation of the city names is all mixed up in this temporary DataFrame. Standardise the strings so that only the first letter is uppercase (e.g. "londON" should become "London".)

In [15]:

```
##Solution
df1[['Depart','Arrival']]=df1[['Depart','Arrival']].apply(lambda x:x.str.capitalize(
df1
```

Out[15]:

	From_To	FlightNumber	RecentDelays	Airline	Depart	Arrival
0	LoNDon_paris	10045	[23, 47]	KLM(!)	London	Paris
1	MAdrid_miLAN	10055	[]	<Air France> (12)	Madrid	Milan
2	londON_StockhOlm	10065	[24, 43, 87]	(British Airways.)	London	Stockholm
3	Budapest_PaRis	10075	[13]	12. Air France	Budapest	Paris
4	Brussels_londOn	10085	[67, 32]	"Swiss Air"	Brussels	London

##Q4.Delete the From_To column from df and attach the temporary DataFrame from the previous questions.

In [22]:

```
#Solution
df1.drop(['From_To'],axis=1,inplace=True)
df1
```

Out[22]:

	FlightNumber	RecentDelays	Airline	Depart	Arrival
0	10045	[23, 47]	KLM(!)	London	Paris
1	10055	[]	<Air France> (12)	Madrid	Milan
2	10065	[24, 43, 87]	(British Airways.)	London	Stockholm
3	10075	[13]	12. Air France	Budapest	Paris
4	10085	[67, 32]	"Swiss Air"	Brussels	London

##Q5.In the RecentDelays column, the values have been entered into the DataFrame as a list. We would like each first value in its own column, each

second value in its own column, and so on. If there isn't an Nth value, the value should be NaN. Expand the Series of lists into a DataFrame named delays, rename the columns delay_1, delay_2, etc. and replace the unwanted RecentDelays column in df with delays.

In [25]:

```
##Solution
rec_delay=df1['RecentDelays'].apply(pd.Series)
rec_delay
```

Out[25]:

	0	1	2
0	23.0	47.0	NaN
1	NaN	NaN	NaN
2	24.0	43.0	87.0
3	13.0	NaN	NaN
4	67.0	32.0	NaN

In [27]:

```
rec_delay.columns=['delay_{}'.format(n) for n in range(1,len(rec_delay.columns)+1)]
rec_delay
```

Out[27]:

	delay_1	delay_2	delay_3
0	23.0	47.0	NaN
1	NaN	NaN	NaN
2	24.0	43.0	87.0
3	13.0	NaN	NaN
4	67.0	32.0	NaN

In [31]:

```
df2=df1.join(rec_delay)
df2.drop('RecentDelays',axis=1,inplace=True)
df2
```

Out[31]:

	FlightNumber	Airline	Depart	Arrival	delay_1	delay_2	delay_3
0	10045	KLM(!)	London	Paris	23.0	47.0	NaN
1	10055	<Air France> (12)	Madrid	Milan	NaN	NaN	NaN
2	10065	(British Airways.)	London	Stockholm	24.0	43.0	87.0
3	10075	12. Air France	Budapest	Paris	13.0	NaN	NaN
4	10085	"Swiss Air"	Brussels	London	67.0	32.0	NaN

In [ ]:

```

```