

```
In [2]: #Explora les dades que conté, i queda't únicament amb les columnes que consideris rellevants.

import sqlite3
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv("DelayedFlights.csv")
df

Out[2]:
```

Unnamed: 0	Year	Month	DayofMonth	DayOfWeek	DepTime	CRSDepTime	ArrTime	CRSArrTime	UniqueCarrier	...	TaxiIn	TaxiOut	Cancelled	CancellationCode	Diverted	CarrierDelay	WeatherDelay
0	0	2008	1	3	4	2003.0	1955	2211.0	2225	WN	...	4.0	8.0	0	N	0	NaN
1	1	2008	1	3	4	754.0	735	1002.0	1000	WN	...	5.0	10.0	0	N	0	NaN
2	2	2008	1	3	4	628.0	620	804.0	750	WN	...	3.0	17.0	0	N	0	NaN
3	4	2008	1	3	4	1829.0	1755	1959.0	1925	WN	...	3.0	10.0	0	N	0	2.0
4	5	2008	1	3	4	1940.0	1915	2121.0	2110	WN	...	4.0	10.0	0	N	0	NaN
...
1936753	7009710	2008	12	13	6	1250.0	1220	1617.0	1552	DL	...	9.0	18.0	0	N	0	3.0
1936754	7009717	2008	12	13	6	657.0	600	904.0	749	DL	...	15.0	34.0	0	N	0	0.0
1936755	7009718	2008	12	13	6	1007.0	847	1149.0	1010	DL	...	8.0	32.0	0	N	0	1.0
1936756	7009726	2008	12	13	6	1251.0	1240	1446.0	1437	DL	...	13.0	13.0	0	N	0	NaN
1936757	7009727	2008	12	13	6	1110.0	1103	1413.0	1418	DL	...	8.0	11.0	0	N	0	NaN

1936758 rows x 30 columns

```
In [3]: novaTaula = df.head(10)
df.head()

Out[3]:
```

Unnamed: 0	Year	Month	DayofMonth	DayOfWeek	DepTime	CRSDepTime	ArrTime	CRSArrTime	UniqueCarrier	...	TaxiIn	TaxiOut	Cancelled	CancellationCode	Diverted	CarrierDelay	WeatherDelay
0	0	2008	1	3	4	2003.0	1955	2211.0	2225	WN	...	4.0	8.0	0	N	0	NaN
1	1	2008	1	3	4	754.0	735	1002.0	1000	WN	...	5.0	10.0	0	N	0	NaN
2	2	2008	1	3	4	628.0	620	804.0	750	WN	...	3.0	17.0	0	N	0	NaN
3	4	2008	1	3	4	1829.0	1755	1959.0	1925	WN	...	3.0	10.0	0	N	0	2.0
4	5	2008	1	3	4	1940.0	1915	2121.0	2110	WN	...	4.0	10.0	0	N	0	NaN

5 rows x 30 columns

```
In [4]: subset = df[['Year', 'Month', 'DayofMonth', 'DayOfWeek', 'ArrTime', 'FlightNum', 'Cancelled', 'DepTime']]

subset = subset.drop(range(11,1936758),axis=0)
print(subset)
```

	Year	Month	DayofMonth	DayOfWeek	ArrTime	FlightNum	Cancelled	DepTime
0	2008	1	3	4	2211.0	335	0	2003.0
1	2008	1	3	4	1002.0	3231	0	754.0
2	2008	1	3	4	804.0	448	0	628.0
3	2008	1	3	4	1959.0	3920	0	1829.0
4	2008	1	3	4	2121.0	378	0	1940.0
5	2008	1	3	4	2037.0	509	0	1937.0
6	2008	1	3	4	916.0	100	0	706.0
7	2008	1	3	4	1845.0	1333	0	1644.0
8	2008	1	3	4	1021.0	2272	0	1029.0
9	2008	1	3	4	1640.0	675	0	1452.0
10	2008	1	3	4	940.0	1144	0	754.0

```
In [5]: #Fes un informe complet del data set:..

#Resumeix estadísticament les columnes d'interès
#Troba quantes dades faltants hi ha per columna
#Crea columnes noves (velocitat mitjana del vol, si ha arribat tard o no...)
#Taula de les aerolínies amb més endarreriments acumulats
#Quins són els vols més llargs? I els més endarrerits?
#Etc.

def func_2(column = 'Year'):
    year= df[column].describe()
    print(year)

func_2('Month')
func_2('DayofMonth')
func_2('DayOfWeek')
func_2('ArrTime')
func_2('FlightNum')
func_2('Cancelled')
# Intentaré ficar-ho tot en un dataframe

count      1.936758e+06
mean        6.111106e+00
std          3.482546e+00
min          1.000000e+00
25%          3.000000e+00
50%          6.000000e+00
75%          9.000000e+00
max          1.200000e+01
Name: Month, dtype: float64
count      1.936758e+06
mean        1.575347e+01
std          8.776272e+00
min          1.000000e+00
25%          8.000000e+00
50%          1.600000e+01
75%          2.300000e+01
max          3.100000e+01
Name: DayofMonth, dtype: float64
count      1.936758e+06
mean        3.984827e+00
std          1.995966e+00
min          1.000000e+00
25%          2.000000e+00
50%          4.000000e+00
75%          6.000000e+00
max          7.000000e+00
Name: DayOfWeek, dtype: float64
count      1.929648e+06
mean        1.610141e+03
std          5.481781e+02
min          1.000000e+00
25%          1.316000e+03
50%          1.715000e+03
75%          2.030000e+03
max          2.400000e+03
Name: ArrTime, dtype: float64
count      1.936758e+06
mean        2.184263e+03
std          1.944702e+03
min          1.000000e+00
25%          6.100000e+02
50%          1.543000e+03
75%          3.422000e+03
max          9.742000e+03
Name: FlightNum, dtype: float64
count      1.936758e+06
mean        3.268348e-04
std          1.807562e-02
min          0.000000e+00
25%          0.000000e+00
50%          0.000000e+00
75%          0.000000e+00
max          1.000000e+00
Name: Cancelled, dtype: float64
```

```
In [6]: #Troba quantes dades faltants hi ha per columna
def function(column = 'Year'):

    columna_1 = df[column]
    res = columna_1.mean()
    x = columna_1.fillna(res, inplace=True)
    print(x)

function()
function('Month')
function('DayofMonth')
function('DayOfWeek')
function('ArrTime')
function('FlightNum')
function('Cancelled')
function('CarrierDelay')
```

```
None
None
None
None
None
None
None
None
```

```
In [7]: #Crea columnes noves (velocitat mitjana del vol, si ha arribat tard o no...)

subset.insert(4, "AvSpeed", ['389', '435', '521', '430', '420', '387','554', '398','465', '654', '567'], True )

subset.insert(6, "Delayed", ['Yes', 'No', 'No', 'No', 'No', 'Yes', 'No', 'Yes','No','No','No'], True)

subset.insert(7, "Passengers", ['350', '560', '480', '650', '380', '430', '420', '550', '540', '370','520'],True)

print(subset)
```

	Year	Month	DayofMonth	DayOfWeek	AvSpeed	ArrTime	Delayed	Passengers	\
0	2008	1	3	4	389	2211.0	Yes	350	
1	2008	1	3	4	435	1002.0	No	560	
2	2008	1	3	4	521	804.0	No	480	
3	2008	1	3	4	430	1959.0	No	650	
4	2008	1	3	4	420	2121.0	No	380	
5	2008	1	3	4	387	2037.0	Yes	430	
6	2008	1	3	4	554	916.0	No	420	
7	2008	1	3	4	398	1845.0	Yes	550	
8	2008	1	3	4	465	1021.0	No	540	
9	2008	1	3	4	654	1640.0	No	370	
10	2008	1	3	4	567	940.0	No	520	

	FlightNum	Cancelled	DepTime
0	335	0	2003.0
1	3231	0	754.0
2	448	0	628.0
3	3920	0	1829.0
4	378	0	1940.0
5	509	0	1937.0
6	100	0	706.0
7	1333	0	1644.0
8	2272	0	1029.0
9	675	0	1452.0
10	1144	0	754.0

```
In [8]: #Taula de les aerolínies amb més endarreriments acumulats

newSet = df[['CarrierDelay', 'WeatherDelay', 'NASDelay', 'SecurityDelay', 'LateAircraftDelay']]

ordenarTard = newSet.sort_values(['CarrierDelay', 'WeatherDelay', 'NASDelay','SecurityDelay','LateAircraftDelay'])

newSet = ordenarTard.sum(axis=1)
print(newSet.tail())
```

```
685437      1542.0
1497823     1583.0
1009553     1707.0
839306      1951.0
686014      2453.0
dtype: float64
```

```
In [13]: ##Quins són els vols més llargs? I els més endarrerits?

resta = df['ArrTime'] - df['DepTime']
print(resta)

x = resta.sort_values()
print(x)

x.tail(20)
```

```
0          208.0
1          248.0
2          176.0
3          130.0
4          181.0
...
1936753     367.0
1936754     247.0
1936755     142.0
1936756     195.0
1936757     303.0
Length: 1936758, dtype: float64
1262416    -2396.0
1170372    -2394.0
1087824    -2391.0
284485     -2385.0
1352970    -2379.0
...
1176623     2357.0
1260284     2377.0
1656812     2385.0
1179896     2392.0
1765176     2397.0
Length: 1936758, dtype: float64
194434      1609.140629
287208      2337.000000
1352190      2339.000000
932288       2340.000000
1601846      2342.000000
1352192      2342.000000
1601840      2342.000000
675808       2343.000000
907881       2345.000000
472955       2347.000000
726753       2347.000000
280274       2348.000000
1179548      2348.000000
1175566      2351.000000
1793034      2354.000000
1176623      2357.000000
1260284      2377.000000
1656812      2385.000000
1179896      2392.000000
1765176      2397.000000
dtype: float64
```

```
Out[13]:
```

```
In [57]: subset.to_excel('novaTaula.xlsx')
```