Limpieza de datos

In [1]:
```
import pandas as pd
import os
```

In [2]:
```
mainpath = "/Users/fsanmartin/python-ml-course-master/datasets/"
filename = "titanic/titanic3.csv"
fullpath = os.path.join(mainpath, filename)
```

In [3]:
```
data = pd.read_csv(fullpath)
```

In [4]:
```
data.head()
```

Out[4]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embarke
0	1	1	Allen, Miss. Elisabeth Walton	female	29.0000	0	0	24160	211.3375	B5	
1	1	1	Allison, Master. Hudson Trevor	male	0.9167	1	2	113781	151.5500	C22 C26	
2	1	0	Allison, Miss. Helen Loraine	female	2.0000	1	2	113781	151.5500	C22 C26	
3	1	0	Allison, Mr. Hudson Joshua Creighton	male	30.0000	1	2	113781	151.5500	C22 C26	
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	25.0000	1	2	113781	151.5500	C22 C26	

In [5]:
```
data2 = pd.read_csv('/Users/fsanmartin/python-ml-course-master/datasets/customer-churn-model/Customer Churn Model.txt')
```

In [6]:
```
data2.head()
```

Out[6]:

	State	Account Length	Area Code	Phone	Int'l Plan	VMail Plan	VMail Message	Day Mins	Day Calls	Day Charge	...	Eve Calls	Eve Charge
0	KS	128	415	382-4657	no	yes	25	265.1	110	45.07	...	99	16.78
1	OH	107	415	371-7191	no	yes	26	161.6	123	27.47	...	103	16.62
2	NJ	137	415	358-1921	no	no	0	243.4	114	41.38	...	110	10.30
3	OH	84	408	375-9999	yes	no	0	299.4	71	50.90	...	88	5.26
4	OK	75	415	330-6626	yes	no	0	166.7	113	28.34	...	122	12.61

5 rows × 21 columns

In [7]:
```
data2.columns
```

Out[7]:
```
Index(['State', 'Account Length', 'Area Code', 'Phone', 'Int'l Plan',
       'VMail Plan', 'VMail Message', 'Day Mins', 'Day Calls', 'Day Charge',
       'Eve Mins', 'Eve Calls', 'Eve Charge', 'Night Mins', 'Night Calls',
       'Night Charge', 'Intl Mins', 'Intl Calls', 'Intl Charge',
       'CustServ Calls', 'Churn?'],
      dtype='object')
```

Carga de datos a través de la función open

In [8]:
```
def df_via_open(path, sep=','):
    '''
    Esta función sirve para crear un dataframe a través de la lectura de un ar
chivo línea a línea.
    Se asume que el nombre de las columnas viene en la primera línea del archi
vo.

    df_via_open(path, sep=',')

    Entradas:

    path = directorio del archivo a cargar
    sep = separador o delimitador entre los datos, por defecto se deja en la c
oma (,).

    Salida:

    Dataframe del archivo ingresado

    '''


    # Se abre el archivo en modo lectura y se almacena en el objeto "data"
    data = open(path, 'r')

    # Seleccionar la primera línea del archivo y convertirla en una lista de
 "n" elementos
    # donde cada elemento representa el nombre de la columna
    columnas = data.readline().strip().split(sep)

    # También se cuentan la cantidad de columnas y se guarda en el objeto "lar
go_columnas"
    largo_columnas = len(columnas)

    # Se inicia un contador y un diccionario ("contador" y "main_dict" respect
ivamente)
    contador = 0
    main_dict = {}

    # Se agrega al diccionario vacío, las columnas obtenidas
    for col in columnas:

        main_dict[col] = []

    # Se realiza un ciclo donde se leerá cada línea del archivo

    for linea in data:

        # Para cada línea se convierte en una lista de "n" elementos
        # cada elemento representa el valor de la variable en la posición i
        values = linea.strip().split(",")

        # Se realiza un recorrido de las posiciones (índices) de cada elemento
perteneciente
        # a la lista de columnas.
        for i in range(len(columnas)):
```

```
            # Se agrega al diccionario el nombre de la columna con su valor re
spectivo
            # asociado a la línea que se está recorriendo
            main_dict[columnas[i]].append(values[i])

        # Se aumenta el contador en 1 por cada línea recorrida
        contador += 1

    # Se transforma el diccionario obtenido a un dataframe
    dataframe = pd.DataFrame(main_dict)

    # Se retorna como resultado el dataframe
    return dataframe
```

In [9]:
```
df = df_via_open('/Users/fsanmartin/python-ml-course-master/datasets/customer-
churn-model/Customer Churn Model.txt')
df.head()
```

Out[9]:

	State	Account Length	Area Code	Phone	Int'l Plan	VMail Plan	VMail Message	Day Mins	Day Calls	Day Charge	...	Eve Calls
0	KS	128	415	382-4657	no	yes	25	265.100000	110	45.070000	...	99
1	OH	107	415	371-7191	no	yes	26	161.600000	123	27.470000	...	103
2	NJ	137	415	358-1921	no	no	0	243.400000	114	41.380000	...	110
3	OH	84	408	375-9999	yes	no	0	299.400000	71	50.900000	...	88
4	OK	75	415	330-6626	yes	no	0	166.700000	113	28.340000	...	122

5 rows × 21 columns

Lectura y escritura de ficheros

In [10]:
```
# Creamos un objeto que contiene el archivo existente de entrada: "infile"
infile = '/Users/fsanmartin/python-ml-course-master/datasets/customer-churn-mo
del/Customer Churn Model.txt'

# Creamos un objeto que contendrá el archivo de salida: "outfile"
outfile = '/Users/fsanmartin/python-ml-course-master/datasets/customer-churn-m
odel/Tab Customer Churn Model.txt'
```

In [11]:
```
# Se abre en modo lectura el archivo de entrada
with open(infile, 'r') as infile1:

    # En modo escritura, abrimos el archvivo de salida
    with open(outfile, 'w') as outfile1:

        # Para cada línea del archivo de entrada
        for line in infile1:

            # Estraemos los "n" elementos de la línea
            fields = line.strip().split(',')

            # Escribimos en el archivo de salida cada elemento de la línea sep
arado por un tabulador

            outfile1.write('\t'.join(fields))

            # Agregamos un saltó de línea al terminar de escribir los elemento
s de la línea.
            outfile1.write('\n')

# Abrimos el archivo de salida como un dataframe, ocupando como separador la t
abulación ('\t')
df4 = pd.read_csv(outfile, sep='\t')

#Revisamos los primeros 5 registros
df4.head()
```

Out[11]:

	State	Account Length	Area Code	Phone	Int'l Plan	VMail Plan	VMail Message	Day Mins	Day Calls	Day Charge	...	Eve Calls	Eve Charge
0	KS	128	415	382-4657	no	yes	25	265.1	110	45.07	...	99	16.78
1	OH	107	415	371-7191	no	yes	26	161.6	123	27.47	...	103	16.62
2	NJ	137	415	358-1921	no	no	0	243.4	114	41.38	...	110	10.30
3	OH	84	408	375-9999	yes	no	0	299.4	71	50.90	...	88	5.26
4	OK	75	415	330-6626	yes	no	0	166.7	113	28.34	...	122	12.61

5 rows × 21 columns

Leer datos desde una URL

In [12]:
```
medals_url = "http://winterolympicsmedals.com/medals.csv"
```

In [13]:
```
medals_data = pd.read_csv(medals_url)
medals_data.head()
```

Out[13]:

	Year	City	Sport	Discipline	NOC	Event	Event gender	Medal
0	1924	Chamonix	Skating	Figure skating	AUT	individual	M	Silver
1	1924	Chamonix	Skating	Figure skating	AUT	individual	W	Gold
2	1924	Chamonix	Skating	Figure skating	AUT	pairs	X	Gold
3	1924	Chamonix	Bobsleigh	Bobsleigh	BEL	four-man	M	Bronze
4	1924	Chamonix	Ice Hockey	Ice Hockey	CAN	ice hockey	M	Gold

In [ ]:
```

```

In [ ]:
```

```

Ficheros XLS y XLSX

In [14]:
```
mainpath = "/Users/fsanmartin/python-ml-course-master/datasets/"
filename = "titanic/titanic3.xls"
fullpath = os.path.join(mainpath, filename)
titanic2 = pd.read_excel(fullpath, 'titanic3')
titanic2.head(1)
```

Out[14]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embarked	bo
0	1	1	Allen, Miss. Elisabeth Walton	female	29.0	0	0	24160	211.3375	B5	S	

Resumen de datos: dimensiones y estructuras

In [15]:
```
data.head()
```

Out[15]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embarke
0	1	1	Allen, Miss. Elisabeth Walton	female	29.0000	0	0	24160	211.3375	B5	
1	1	1	Allison, Master. Hudson Trevor	male	0.9167	1	2	113781	151.5500	C22 C26	
2	1	0	Allison, Miss. Helen Loraine	female	2.0000	1	2	113781	151.5500	C22 C26	
3	1	0	Allison, Mr. Hudson Joshua Creighton	male	30.0000	1	2	113781	151.5500	C22 C26	
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	25.0000	1	2	113781	151.5500	C22 C26	

In [16]:
```
data.shape
```

Out[16]:
```
(1309, 14)
```

In [17]:
```
data.tail()
```

Out[17]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embark
1304	3	0	Zabour, Miss. Hileni	female	14.5	1	0	2665	14.4542	NaN	
1305	3	0	Zabour, Miss. Thamine	female	NaN	1	0	2665	14.4542	NaN	
1306	3	0	Zakarian, Mr. Mapriededer	male	26.5	0	0	2656	7.2250	NaN	
1307	3	0	Zakarian, Mr. Ortin	male	27.0	0	0	2670	7.2250	NaN	
1308	3	0	Zimmerman, Mr. Leo	male	29.0	0	0	315082	7.8750	NaN	

In [18]:
```
data.columns.values
```

Out[18]:
```
array(['pclass', 'survived', 'name', 'sex', 'age', 'sibsp', 'parch',
       'ticket', 'fare', 'cabin', 'embarked', 'boat', 'body', 'home.dest'],
      dtype=object)
```

Resumen básico de las variables numéricas

In [19]:
```
data.describe()
```

Out[19]:

	pclass	survived	age	sibsp	parch	fare	bod
count	1309.000000	1309.000000	1046.000000	1309.000000	1309.000000	1308.000000	121.00000
mean	2.294882	0.381971	29.881135	0.498854	0.385027	33.295479	160.80991
std	0.837836	0.486055	14.413500	1.041658	0.865560	51.758668	97.69692
min	1.000000	0.000000	0.166700	0.000000	0.000000	0.000000	1.00000
25%	2.000000	0.000000	21.000000	0.000000	0.000000	7.895800	72.00000
50%	3.000000	0.000000	28.000000	0.000000	0.000000	14.454200	155.00000
75%	3.000000	1.000000	39.000000	1.000000	0.000000	31.275000	256.00000
max	3.000000	1.000000	80.000000	8.000000	9.000000	512.329200	328.00000

In [20]:
```
data.dtypes
```

Out[20]:
```
pclass         int64
survived       int64
name          object
sex           object
age          float64
sibsp          int64
parch          int64
ticket        object
fare         float64
cabin         object
embarked      object
boat          object
body         float64
home.dest     object
dtype: object
```

Datos perdidos

In [21]:
```
data['body'].isnull().values.sum()
```

Out[21]:
```
1188
```

Los valores que faltan en un data set pueden venir por dos razones:

- Extracción de los datos
- Recolección de los datos

¿Qué hacer con los valores perdidos?

1. Borrado de valores que faltan

1.1.- Borrar las filas con datos perdidos

1.2.- Borrar la columna

In [22]:
```
#Borrar la fila, solo si todas las columnas son valores perdidos
data.dropna(axis=0, how="all")
```

Out[22]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabi
0	1	1	Allen, Miss. Elisabeth Walton	female	29.0000	0	0	24160	211.3375	B
1	1	1	Allison, Master. Hudson Trevor	male	0.9167	1	2	113781	151.5500	C2 C2
2	1	0	Allison, Miss. Helen Loraine	female	2.0000	1	2	113781	151.5500	C2 C2
3	1	0	Allison, Mr. Hudson Joshua Creighton	male	30.0000	1	2	113781	151.5500	C2 C2
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	25.0000	1	2	113781	151.5500	C2 C2
5	1	1	Anderson, Mr. Harry	male	48.0000	0	0	19952	26.5500	E1
6	1	1	Andrews, Miss. Kornelia Theodosia	female	63.0000	1	0	13502	77.9583	D
7	1	0	Andrews, Mr. Thomas Jr	male	39.0000	0	0	112050	0.0000	A3
8	1	1	Appleton, Mrs. Edward Dale (Charlotte Lamson)	female	53.0000	2	0	11769	51.4792	C10
9	1	0	Artagaveytia, Mr. Ramon	male	71.0000	0	0	PC 17609	49.5042	NaI
10	1	0	Astor, Col. John Jacob	male	47.0000	1	0	PC 17757	227.5250	C6 C6
11	1	1	Astor, Mrs. John Jacob (Madeleine Talmadge Force)	female	18.0000	1	0	PC 17757	227.5250	C6 C6
12	1	1	Aubart, Mme. Leontine Pauline	female	24.0000	0	0	PC 17477	69.3000	B3
13	1	1	Barber, Miss. Ellen "Nellie"	female	26.0000	0	0	19877	78.8500	NaI
14	1	1	Barkworth, Mr. Algernon Henry Wilson	male	80.0000	0	0	27042	30.0000	A2
15	1	0	Baumann, Mr. John D	male	NaN	0	0	PC 17318	25.9250	NaI

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabi
16	1	0	Baxter, Mr. Quigg Edmond	male	24.0000	0	1	PC 17558	247.5208	B5 B6
17	1	1	Baxter, Mrs. James (Helene DeLaudeniere Chaput)	female	50.0000	0	1	PC 17558	247.5208	B5 B6
18	1	1	Bazzani, Miss. Albina	female	32.0000	0	0	11813	76.2917	D1
19	1	0	Beattie, Mr. Thomson	male	36.0000	0	0	13050	75.2417	C
20	1	1	Beckwith, Mr. Richard Leonard	male	37.0000	1	1	11751	52.5542	D3
21	1	1	Beckwith, Mrs. Richard Leonard (Sallie Monypeny)	female	47.0000	1	1	11751	52.5542	D3
22	1	1	Behr, Mr. Karl Howell	male	26.0000	0	0	111369	30.0000	C14
23	1	1	Bidois, Miss. Rosalie	female	42.0000	0	0	PC 17757	227.5250	NaI
24	1	1	Bird, Miss. Ellen	female	29.0000	0	0	PC 17483	221.7792	C9
25	1	0	Birnbaum, Mr. Jakob	male	25.0000	0	0	13905	26.0000	NaI
26	1	1	Bishop, Mr. Dickinson H	male	25.0000	1	0	11967	91.0792	B4
27	1	1	Bishop, Mrs. Dickinson H (Helen Walton)	female	19.0000	1	0	11967	91.0792	B4
28	1	1	Bissette, Miss. Amelia	female	35.0000	0	0	PC 17760	135.6333	C9
29	1	1	Bjornstrom-Steffansson, Mr. Mauritz Hakan	male	28.0000	0	0	110564	26.5500	C5
...	...	...	...	...	...	...	...	...	...	.
1279	3	0	Vestrom, Miss. Hulda Amanda Adolfina	female	14.0000	0	0	350406	7.8542	NaI
1280	3	0	Vovk, Mr. Janko	male	22.0000	0	0	349252	7.8958	NaI
1281	3	0	Waelens, Mr. Achille	male	22.0000	0	0	345767	9.0000	NaI

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabi
1282	3	0	Ware, Mr. Frederick	male	NaN	0	0	359309	8.0500	Nal
1283	3	0	Warren, Mr. Charles William	male	NaN	0	0	C.A. 49867	7.5500	Nal
1284	3	0	Webber, Mr. James	male	NaN	0	0	SOTON/OQ 3101316	8.0500	Nal
1285	3	0	Wenzel, Mr. Linhart	male	32.5000	0	0	345775	9.5000	Nal
1286	3	1	Whabee, Mrs. George Joseph (Shawneene Abi-Saab)	female	38.0000	0	0	2688	7.2292	Nal
1287	3	0	Widegren, Mr. Carl/Charles Peter	male	51.0000	0	0	347064	7.7500	Nal
1288	3	0	Wiklund, Mr. Jakob Alfred	male	18.0000	1	0	3101267	6.4958	Nal
1289	3	0	Wiklund, Mr. Karl Johan	male	21.0000	1	0	3101266	6.4958	Nal
1290	3	1	Wilkes, Mrs. James (Ellen Needs)	female	47.0000	1	0	363272	7.0000	Nal
1291	3	0	Willer, Mr. Aaron ("Abi Weller")	male	NaN	0	0	3410	8.7125	Nal
1292	3	0	Willey, Mr. Edward	male	NaN	0	0	S.O./P.P. 751	7.5500	Nal
1293	3	0	Williams, Mr. Howard Hugh "Harry"	male	NaN	0	0	A/5 2466	8.0500	Nal
1294	3	0	Williams, Mr. Leslie	male	28.5000	0	0	54636	16.1000	Nal
1295	3	0	Windelov, Mr. Einar	male	21.0000	0	0	SOTON/OQ 3101317	7.2500	Nal
1296	3	0	Wirz, Mr. Albert	male	27.0000	0	0	315154	8.6625	Nal
1297	3	0	Wiseman, Mr. Phillippe	male	NaN	0	0	A/4. 34244	7.2500	Nal
1298	3	0	Wittevrongel, Mr. Camille	male	36.0000	0	0	345771	9.5000	Nal
1299	3	0	Yasbeck, Mr. Antoni	male	27.0000	1	0	2659	14.4542	Nal
1300	3	1	Yasbeck, Mrs. Antoni (Selini Alexander)	female	15.0000	1	0	2659	14.4542	Nal
1301	3	0	Youseff, Mr. Gerious	male	45.5000	0	0	2628	7.2250	Nal

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabi
1302	3	0	Yousif, Mr. Wazli	male	NaN	0	0	2647	7.2250	Nal
1303	3	0	Yousseff, Mr. Gerious	male	NaN	0	0	2627	14.4583	Nal
1304	3	0	Zabour, Miss. Hileni	female	14.5000	1	0	2665	14.4542	Nal
1305	3	0	Zabour, Miss. Thamine	female	NaN	1	0	2665	14.4542	Nal
1306	3	0	Zakarian, Mr. Mapriededer	male	26.5000	0	0	2656	7.2250	Nal
1307	3	0	Zakarian, Mr. Ortin	male	27.0000	0	0	2670	7.2250	Nal
1308	3	0	Zimmerman, Mr. Leo	male	29.0000	0	0	315082	7.8750	Nal

1309 rows × 14 columns

In [23]:
```
data2 = data

#Borrar la fila, si al menos una de las columnas es un valor perdido
data2.dropna(axis=0, how="any")
```

Out[23]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embarked	boat	body	h

¿Qué ocurrió? Pues, sólo existen filas que tienen al menos un valor perdido en sus columnas

Cómputo de los valores faltantes

In [24]:
```
data3 = data


# Rellenar los valores perdidos con un 0
data3.fillna(0)
```

Out[24]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabi
0	1	1	Allen, Miss. Elisabeth Walton	female	29.0000	0	0	24160	211.3375	B
1	1	1	Allison, Master. Hudson Trevor	male	0.9167	1	2	113781	151.5500	C2 C2
2	1	0	Allison, Miss. Helen Loraine	female	2.0000	1	2	113781	151.5500	C2 C2
3	1	0	Allison, Mr. Hudson Joshua Creighton	male	30.0000	1	2	113781	151.5500	C2 C2
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	25.0000	1	2	113781	151.5500	C2 C2
5	1	1	Anderson, Mr. Harry	male	48.0000	0	0	19952	26.5500	E1
6	1	1	Andrews, Miss. Kornelia Theodosia	female	63.0000	1	0	13502	77.9583	D
7	1	0	Andrews, Mr. Thomas Jr	male	39.0000	0	0	112050	0.0000	A3
8	1	1	Appleton, Mrs. Edward Dale (Charlotte Lamson)	female	53.0000	2	0	11769	51.4792	C10
9	1	0	Artagaveytia, Mr. Ramon	male	71.0000	0	0	PC 17609	49.5042	
10	1	0	Astor, Col. John Jacob	male	47.0000	1	0	PC 17757	227.5250	C6 C6
11	1	1	Astor, Mrs. John Jacob (Madeleine Talmadge Force)	female	18.0000	1	0	PC 17757	227.5250	C6 C6
12	1	1	Aubart, Mme. Leontine Pauline	female	24.0000	0	0	PC 17477	69.3000	B3
13	1	1	Barber, Miss. Ellen "Nellie"	female	26.0000	0	0	19877	78.8500	
14	1	1	Barkworth, Mr. Algernon Henry Wilson	male	80.0000	0	0	27042	30.0000	A2
15	1	0	Baumann, Mr. John D	male	0.0000	0	0	PC 17318	25.9250	

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabi
16	1	0	Baxter, Mr. Quigg Edmond	male	24.0000	0	1	PC 17558	247.5208	B5 B6
17	1	1	Baxter, Mrs. James (Helene DeLaudeniere Chaput)	female	50.0000	0	1	PC 17558	247.5208	B5 B6
18	1	1	Bazzani, Miss. Albina	female	32.0000	0	0	11813	76.2917	D1
19	1	0	Beattie, Mr. Thomson	male	36.0000	0	0	13050	75.2417	C
20	1	1	Beckwith, Mr. Richard Leonard	male	37.0000	1	1	11751	52.5542	D3
21	1	1	Beckwith, Mrs. Richard Leonard (Sallie Monypeny)	female	47.0000	1	1	11751	52.5542	D3
22	1	1	Behr, Mr. Karl Howell	male	26.0000	0	0	111369	30.0000	C14
23	1	1	Bidois, Miss. Rosalie	female	42.0000	0	0	PC 17757	227.5250	
24	1	1	Bird, Miss. Ellen	female	29.0000	0	0	PC 17483	221.7792	C9
25	1	0	Birnbaum, Mr. Jakob	male	25.0000	0	0	13905	26.0000	
26	1	1	Bishop, Mr. Dickinson H	male	25.0000	1	0	11967	91.0792	B4
27	1	1	Bishop, Mrs. Dickinson H (Helen Walton)	female	19.0000	1	0	11967	91.0792	B4
28	1	1	Bissette, Miss. Amelia	female	35.0000	0	0	PC 17760	135.6333	C9
29	1	1	Bjornstrom-Steffansson, Mr. Mauritz Hakan	male	28.0000	0	0	110564	26.5500	C5
...	...	...	...	...	...	...	...	...	...	.
1279	3	0	Vestrom, Miss. Hulda Amanda Adolfina	female	14.0000	0	0	350406	7.8542	
1280	3	0	Vovk, Mr. Janko	male	22.0000	0	0	349252	7.8958	
1281	3	0	Waelens, Mr. Achille	male	22.0000	0	0	345767	9.0000	

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabi
1282	3	0	Ware, Mr. Frederick	male	0.0000	0	0	359309	8.0500	
1283	3	0	Warren, Mr. Charles William	male	0.0000	0	0	C.A. 49867	7.5500	
1284	3	0	Webber, Mr. James	male	0.0000	0	0	SOTON/OQ 3101316	8.0500	
1285	3	0	Wenzel, Mr. Linhart	male	32.5000	0	0	345775	9.5000	
1286	3	1	Whabee, Mrs. George Joseph (Shawneene Abi-Saab)	female	38.0000	0	0	2688	7.2292	
1287	3	0	Widegren, Mr. Carl/Charles Peter	male	51.0000	0	0	347064	7.7500	
1288	3	0	Wiklund, Mr. Jakob Alfred	male	18.0000	1	0	3101267	6.4958	
1289	3	0	Wiklund, Mr. Karl Johan	male	21.0000	1	0	3101266	6.4958	
1290	3	1	Wilkes, Mrs. James (Ellen Needs)	female	47.0000	1	0	363272	7.0000	
1291	3	0	Willer, Mr. Aaron ("Abi Weller")	male	0.0000	0	0	3410	8.7125	
1292	3	0	Willey, Mr. Edward	male	0.0000	0	0	S.O./P.P. 751	7.5500	
1293	3	0	Williams, Mr. Howard Hugh "Harry"	male	0.0000	0	0	A/5 2466	8.0500	
1294	3	0	Williams, Mr. Leslie	male	28.5000	0	0	54636	16.1000	
1295	3	0	Windelov, Mr. Einar	male	21.0000	0	0	SOTON/OQ 3101317	7.2500	
1296	3	0	Wirz, Mr. Albert	male	27.0000	0	0	315154	8.6625	
1297	3	0	Wiseman, Mr. Phillippe	male	0.0000	0	0	A/4. 34244	7.2500	
1298	3	0	Wittevrongel, Mr. Camille	male	36.0000	0	0	345771	9.5000	
1299	3	0	Yasbeck, Mr. Antoni	male	27.0000	1	0	2659	14.4542	
1300	3	1	Yasbeck, Mrs. Antoni (Selini Alexander)	female	15.0000	1	0	2659	14.4542	
1301	3	0	Youseff, Mr. Gerious	male	45.5000	0	0	2628	7.2250	

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabi
1302	3	0	Yousif, Mr. Wazli	male	0.0000	0	0	2647	7.2250	
1303	3	0	Yousseff, Mr. Gerious	male	0.0000	0	0	2627	14.4583	
1304	3	0	Zabour, Miss. Hileni	female	14.5000	1	0	2665	14.4542	
1305	3	0	Zabour, Miss. Thamine	female	0.0000	1	0	2665	14.4542	
1306	3	0	Zakarian, Mr. Mapriededer	male	26.5000	0	0	2656	7.2250	
1307	3	0	Zakarian, Mr. Ortin	male	27.0000	0	0	2670	7.2250	
1308	3	0	Zimmerman, Mr. Leo	male	29.0000	0	0	315082	7.8750	

1309 rows × 14 columns

In [25]:
```
data4 = data
# Rellenar los valores perdidos con un string
data4.fillna('Desconocido')
```

Out[25]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare
0	1	1	Allen, Miss. Elisabeth Walton	female	29	0	0	24160	211.338
1	1	1	Allison, Master. Hudson Trevor	male	0.9167	1	2	113781	151.55
2	1	0	Allison, Miss. Helen Loraine	female	2	1	2	113781	151.55
3	1	0	Allison, Mr. Hudson Joshua Creighton	male	30	1	2	113781	151.55
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	25	1	2	113781	151.55
5	1	1	Anderson, Mr. Harry	male	48	0	0	19952	26.55
6	1	1	Andrews, Miss. Kornelia Theodosia	female	63	1	0	13502	77.9583
7	1	0	Andrews, Mr. Thomas Jr	male	39	0	0	112050	0
8	1	1	Appleton, Mrs. Edward Dale (Charlotte Lamson)	female	53	2	0	11769	51.4792
9	1	0	Artagaveytia, Mr. Ramon	male	71	0	0	PC 17609	49.5042
10	1	0	Astor, Col. John Jacob	male	47	1	0	PC 17757	227.525
11	1	1	Astor, Mrs. John Jacob (Madeleine Talmadge Force)	female	18	1	0	PC 17757	227.525
12	1	1	Aubart, Mme. Leontine Pauline	female	24	0	0	PC 17477	69.3
13	1	1	Barber, Miss. Ellen "Nellie"	female	26	0	0	19877	78.85
14	1	1	Barkworth, Mr. Algernon Henry Wilson	male	80	0	0	27042	30
15	1	0	Baumann, Mr. John D	male	Desconocido	0	0	PC 17318	25.925

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	
16	1	0	Baxter, Mr. Quigg Edmond	male	24	0	1	PC 17558	247.521	
17	1	1	Baxter, Mrs. James (Helene DeLaudeniere Chaput)	female	50	0	1	PC 17558	247.521	
18	1	1	Bazzani, Miss. Albina	female	32	0	0	11813	76.2917	
19	1	0	Beattie, Mr. Thomson	male	36	0	0	13050	75.2417	
20	1	1	Beckwith, Mr. Richard Leonard	male	37	1	1	11751	52.5542	
21	1	1	Beckwith, Mrs. Richard Leonard (Sallie Monypeny)	female	47	1	1	11751	52.5542	
22	1	1	Behr, Mr. Karl Howell	male	26	0	0	111369	30	
23	1	1	Bidois, Miss. Rosalie	female	42	0	0	PC 17757	227.525	D
24	1	1	Bird, Miss. Ellen	female	29	0	0	PC 17483	221.779	
25	1	0	Birnbaum, Mr. Jakob	male	25	0	0	13905	26	D
26	1	1	Bishop, Mr. Dickinson H	male	25	1	0	11967	91.0792	
27	1	1	Bishop, Mrs. Dickinson H (Helen Walton)	female	19	1	0	11967	91.0792	
28	1	1	Bissette, Miss. Amelia	female	35	0	0	PC 17760	135.633	
29	1	1	Bjornstrom-Steffansson, Mr. Mauritz Hakan	male	28	0	0	110564	26.55	
...	...	...	...	...	...	...	...	...	...	
1279	3	0	Vestrom, Miss. Hulda Amanda Adolfina	female	14	0	0	350406	7.8542	D
1280	3	0	Vovk, Mr. Janko	male	22	0	0	349252	7.8958	D
1281	3	0	Waelens, Mr. Achille	male	22	0	0	345767	9	D

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	
1282	3	0	Ware, Mr. Frederick	male	Desconocido	0	0	359309	8.05	D
1283	3	0	Warren, Mr. Charles William	male	Desconocido	0	0	C.A. 49867	7.55	D
1284	3	0	Webber, Mr. James	male	Desconocido	0	0	SOTON/OQ 3101316	8.05	D
1285	3	0	Wenzel, Mr. Linhart	male	32.5	0	0	345775	9.5	D
1286	3	1	Whabee, Mrs. George Joseph (Shawneene Abi-Saab)	female	38	0	0	2688	7.2292	D
1287	3	0	Widegren, Mr. Carl/Charles Peter	male	51	0	0	347064	7.75	D
1288	3	0	Wiklund, Mr. Jakob Alfred	male	18	1	0	3101267	6.4958	D
1289	3	0	Wiklund, Mr. Karl Johan	male	21	1	0	3101266	6.4958	D
1290	3	1	Wilkes, Mrs. James (Ellen Needs)	female	47	1	0	363272	7	D
1291	3	0	Willer, Mr. Aaron ("Abi Weller")	male	Desconocido	0	0	3410	8.7125	D
1292	3	0	Willey, Mr. Edward	male	Desconocido	0	0	S.O./P.P. 751	7.55	D
1293	3	0	Williams, Mr. Howard Hugh "Harry"	male	Desconocido	0	0	A/5 2466	8.05	D
1294	3	0	Williams, Mr. Leslie	male	28.5	0	0	54636	16.1	D
1295	3	0	Windelov, Mr. Einar	male	21	0	0	SOTON/OQ 3101317	7.25	D
1296	3	0	Wirz, Mr. Albert	male	27	0	0	315154	8.6625	D
1297	3	0	Wiseman, Mr. Phillippe	male	Desconocido	0	0	A/4. 34244	7.25	D
1298	3	0	Wittevrongel, Mr. Camille	male	36	0	0	345771	9.5	D
1299	3	0	Yasbeck, Mr. Antoni	male	27	1	0	2659	14.4542	D
1300	3	1	Yasbeck, Mrs. Antoni (Selini Alexander)	female	15	1	0	2659	14.4542	D
1301	3	0	Youseff, Mr. Gerious	male	45.5	0	0	2628	7.225	D

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	
1302	3	0	Yousif, Mr. Wazli	male	Desconocido	0	0	2647	7.225	D
1303	3	0	Yousseff, Mr. Gerious	male	Desconocido	0	0	2627	14.4583	D
1304	3	0	Zabour, Miss. Hileni	female	14.5	1	0	2665	14.4542	D
1305	3	0	Zabour, Miss. Thamine	female	Desconocido	1	0	2665	14.4542	D
1306	3	0	Zakarian, Mr. Mapriededer	male	26.5	0	0	2656	7.225	D
1307	3	0	Zakarian, Mr. Ortin	male	27	0	0	2670	7.225	D
1308	3	0	Zimmerman, Mr. Leo	male	29	0	0	315082	7.875	D

1309 rows × 14 columns

In [26]:
```
data5 = data

# Rellenar los valores perdidos con un string o 0 dependiendo del tipo de dato de la columna

data5['body'] = data5['body'].fillna(0)
data5['home.dest'] = data5['home.dest'].fillna('Desconocido')
data5.tail()
```

Out[26]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embark
1304	3	0	Zabour, Miss. Hileni	female	14.5	1	0	2665	14.4542	NaN	
1305	3	0	Zabour, Miss. Thamine	female	NaN	1	0	2665	14.4542	NaN	
1306	3	0	Zakarian, Mr. Mapriededer	male	26.5	0	0	2656	7.2250	NaN	
1307	3	0	Zakarian, Mr. Ortin	male	27.0	0	0	2670	7.2250	NaN	
1308	3	0	Zimmerman, Mr. Leo	male	29.0	0	0	315082	7.8750	NaN	

In [27]:
```
# Rellenar los valores perdidos con el promedio para una variable numérica

data5['age'] = data5['age'].fillna(round(data['age'].mean(),0))
data5.head(10)
```

Out[27]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	emba
0	1	1	Allen, Miss. Elisabeth Walton	female	29.0000	0	0	24160	211.3375	B5	
1	1	1	Allison, Master. Hudson Trevor	male	0.9167	1	2	113781	151.5500	C22 C26	
2	1	0	Allison, Miss. Helen Loraine	female	2.0000	1	2	113781	151.5500	C22 C26	
3	1	0	Allison, Mr. Hudson Joshua Creighton	male	30.0000	1	2	113781	151.5500	C22 C26	
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	25.0000	1	2	113781	151.5500	C22 C26	
5	1	1	Anderson, Mr. Harry	male	48.0000	0	0	19952	26.5500	E12	
6	1	1	Andrews, Miss. Kornelia Theodosia	female	63.0000	1	0	13502	77.9583	D7	
7	1	0	Andrews, Mr. Thomas Jr	male	39.0000	0	0	112050	0.0000	A36	
8	1	1	Appleton, Mrs. Edward Dale (Charlotte Lamson)	female	53.0000	2	0	11769	51.4792	C101	
9	1	0	Artagaveytia, Mr. Ramon	male	71.0000	0	0	PC 17609	49.5042	NaN	

In [28]:
```
# Rellenar los valores perdidos con el valor siguiente conocido

data4['age'].fillna(method='ffill')
```

Out[28]:

0	29.0000
1	0.9167
2	2.0000
3	30.0000
4	25.0000
5	48.0000
6	63.0000
7	39.0000
8	53.0000
9	71.0000
10	47.0000
11	18.0000
12	24.0000
13	26.0000
14	80.0000
15	30.0000
16	24.0000
17	50.0000
18	32.0000
19	36.0000
20	37.0000
21	47.0000
22	26.0000
23	42.0000
24	29.0000
25	25.0000
26	25.0000
27	19.0000
28	35.0000
29	28.0000
...	...
1279	14.0000
1280	22.0000
1281	22.0000
1282	30.0000
1283	30.0000
1284	30.0000
1285	32.5000
1286	38.0000
1287	51.0000
1288	18.0000
1289	21.0000
1290	47.0000
1291	30.0000
1292	30.0000
1293	30.0000
1294	28.5000
1295	21.0000
1296	27.0000
1297	30.0000
1298	36.0000
1299	27.0000
1300	15.0000
1301	45.5000
1302	30.0000
1303	30.0000
1304	14.5000

```
1305    30.0000
1306    26.5000
1307    27.0000
1308    29.0000
Name: age, Length: 1309, dtype: float64
```

In [29]:
```
# Rellenar los valores perdidos con el valor anterior conocido

data4['age'].fillna(method='bfill')
```

Out[29]:

0	29.0000
1	0.9167
2	2.0000
3	30.0000
4	25.0000
5	48.0000
6	63.0000
7	39.0000
8	53.0000
9	71.0000
10	47.0000
11	18.0000
12	24.0000
13	26.0000
14	80.0000
15	30.0000
16	24.0000
17	50.0000
18	32.0000
19	36.0000
20	37.0000
21	47.0000
22	26.0000
23	42.0000
24	29.0000
25	25.0000
26	25.0000
27	19.0000
28	35.0000
29	28.0000
...	...
1279	14.0000
1280	22.0000
1281	22.0000
1282	30.0000
1283	30.0000
1284	30.0000
1285	32.5000
1286	38.0000
1287	51.0000
1288	18.0000
1289	21.0000
1290	47.0000
1291	30.0000
1292	30.0000
1293	30.0000
1294	28.5000
1295	21.0000
1296	27.0000
1297	30.0000
1298	36.0000
1299	27.0000
1300	15.0000
1301	45.5000
1302	30.0000
1303	30.0000
1304	14.5000

```
1305    30.0000
1306    26.5000
1307    27.0000
1308    29.0000
Name: age, Length: 1309, dtype: float64
```

Variables dummy

In [30]:
```
#Uso de get_dummies
#Proceso de generar n columnas a partir de los valores dentro de una columna d
el tipo categórica

dummy_sex = pd.get_dummies(data['sex'], prefix='sex')
dummy_sex.head()
```

Out[30]:

	sex_female	sex_male
0	1	0
1	0	1
2	1	0
3	0	1
4	1	0

In [31]:
```
data['sex'].head()
```

Out[31]:
```
0    female
1      male
2    female
3      male
4    female
Name: sex, dtype: object
```

In [32]:
```
#Se debe eliminar la variable original y agregar la forma dummyzada*
data = data.drop(['sex'], axis=1)
# Se agregan las columnas dummyzadas con concat
pd.concat([data, dummy_sex], axis = 1)
```

Out[32]:

	pclass	survived	name	age	sibsp	parch	ticket	fare	cabin	emba
0	1	1	Allen, Miss. Elisabeth Walton	29.0000	0	0	24160	211.3375	B5	
1	1	1	Allison, Master. Hudson Trevor	0.9167	1	2	113781	151.5500	C22 C26	
2	1	0	Allison, Miss. Helen Loraine	2.0000	1	2	113781	151.5500	C22 C26	
3	1	0	Allison, Mr. Hudson Joshua Creighton	30.0000	1	2	113781	151.5500	C22 C26	
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	25.0000	1	2	113781	151.5500	C22 C26	
5	1	1	Anderson, Mr. Harry	48.0000	0	0	19952	26.5500	E12	
6	1	1	Andrews, Miss. Kornelia Theodosia	63.0000	1	0	13502	77.9583	D7	
7	1	0	Andrews, Mr. Thomas Jr	39.0000	0	0	112050	0.0000	A36	
8	1	1	Appleton, Mrs. Edward Dale (Charlotte Lamson)	53.0000	2	0	11769	51.4792	C101	
9	1	0	Artagaveytia, Mr. Ramon	71.0000	0	0	PC 17609	49.5042	NaN	
10	1	0	Astor, Col. John Jacob	47.0000	1	0	PC 17757	227.5250	C62 C64	
11	1	1	Astor, Mrs. John Jacob (Madeleine Talmadge Force)	18.0000	1	0	PC 17757	227.5250	C62 C64	
12	1	1	Aubart, Mme. Leontine Pauline	24.0000	0	0	PC 17477	69.3000	B35	
13	1	1	Barber, Miss. Ellen "Nellie"	26.0000	0	0	19877	78.8500	NaN	
14	1	1	Barkworth, Mr. Algernon Henry Wilson	80.0000	0	0	27042	30.0000	A23	
15	1	0	Baumann, Mr. John D	30.0000	0	0	PC 17318	25.9250	NaN	

	pclass	survived	name	age	sibsp	parch	ticket	fare	cabin	emba
16	1	0	Baxter, Mr. Quigg Edmond	24.0000	0	1	PC 17558	247.5208	B58 B60	
17	1	1	Baxter, Mrs. James (Helene DeLaudeniere Chaput)	50.0000	0	1	PC 17558	247.5208	B58 B60	
18	1	1	Bazzani, Miss. Albina	32.0000	0	0	11813	76.2917	D15	
19	1	0	Beattie, Mr. Thomson	36.0000	0	0	13050	75.2417	C6	
20	1	1	Beckwith, Mr. Richard Leonard	37.0000	1	1	11751	52.5542	D35	
21	1	1	Beckwith, Mrs. Richard Leonard (Sallie Monypeny)	47.0000	1	1	11751	52.5542	D35	
22	1	1	Behr, Mr. Karl Howell	26.0000	0	0	111369	30.0000	C148	
23	1	1	Bidois, Miss. Rosalie	42.0000	0	0	PC 17757	227.5250	NaN	
24	1	1	Bird, Miss. Ellen	29.0000	0	0	PC 17483	221.7792	C97	
25	1	0	Birnbaum, Mr. Jakob	25.0000	0	0	13905	26.0000	NaN	
26	1	1	Bishop, Mr. Dickinson H	25.0000	1	0	11967	91.0792	B49	
27	1	1	Bishop, Mrs. Dickinson H (Helen Walton)	19.0000	1	0	11967	91.0792	B49	
28	1	1	Bissette, Miss. Amelia	35.0000	0	0	PC 17760	135.6333	C99	
29	1	1	Bjornstrom-Steffansson, Mr. Mauritz Hakan	28.0000	0	0	110564	26.5500	C52	
...	...	...	...	...	...	...	...	...	...	
1279	3	0	Vestrom, Miss. Hulda Amanda Adolfina	14.0000	0	0	350406	7.8542	NaN	
1280	3	0	Vovk, Mr. Janko	22.0000	0	0	349252	7.8958	NaN	
1281	3	0	Waelens, Mr. Achille	22.0000	0	0	345767	9.0000	NaN	

	pclass	survived	name	age	sibsp	parch	ticket	fare	cabin	emba
1282	3	0	Ware, Mr. Frederick	30.0000	0	0	359309	8.0500	NaN	
1283	3	0	Warren, Mr. Charles William	30.0000	0	0	C.A. 49867	7.5500	NaN	
1284	3	0	Webber, Mr. James	30.0000	0	0	SOTON/OQ 3101316	8.0500	NaN	
1285	3	0	Wenzel, Mr. Linhart	32.5000	0	0	345775	9.5000	NaN	
1286	3	1	Whabee, Mrs. George Joseph (Shawneene Abi-Saab)	38.0000	0	0	2688	7.2292	NaN	
1287	3	0	Widegren, Mr. Carl/Charles Peter	51.0000	0	0	347064	7.7500	NaN	
1288	3	0	Wiklund, Mr. Jakob Alfred	18.0000	1	0	3101267	6.4958	NaN	
1289	3	0	Wiklund, Mr. Karl Johan	21.0000	1	0	3101266	6.4958	NaN	
1290	3	1	Wilkes, Mrs. James (Ellen Needs)	47.0000	1	0	363272	7.0000	NaN	
1291	3	0	Willer, Mr. Aaron ("Abi Weller")	30.0000	0	0	3410	8.7125	NaN	
1292	3	0	Willey, Mr. Edward	30.0000	0	0	S.O./P.P. 751	7.5500	NaN	
1293	3	0	Williams, Mr. Howard Hugh "Harry"	30.0000	0	0	A/5 2466	8.0500	NaN	
1294	3	0	Williams, Mr. Leslie	28.5000	0	0	54636	16.1000	NaN	
1295	3	0	Windelov, Mr. Einar	21.0000	0	0	SOTON/OQ 3101317	7.2500	NaN	
1296	3	0	Wirz, Mr. Albert	27.0000	0	0	315154	8.6625	NaN	
1297	3	0	Wiseman, Mr. Phillippe	30.0000	0	0	A/4. 34244	7.2500	NaN	
1298	3	0	Wittevrongel, Mr. Camille	36.0000	0	0	345771	9.5000	NaN	
1299	3	0	Yasbeck, Mr. Antoni	27.0000	1	0	2659	14.4542	NaN	
1300	3	1	Yasbeck, Mrs. Antoni (Selini Alexander)	15.0000	1	0	2659	14.4542	NaN	
1301	3	0	Youseff, Mr. Gerious	45.5000	0	0	2628	7.2250	NaN	

	pclass	survived	name	age	sibsp	parch	ticket	fare	cabin	emba
1302	3	0	Yousif, Mr. Wazli	30.0000	0	0	2647	7.2250	NaN	
1303	3	0	Yousseff, Mr. Gerious	30.0000	0	0	2627	14.4583	NaN	
1304	3	0	Zabour, Miss. Hileni	14.5000	1	0	2665	14.4542	NaN	
1305	3	0	Zabour, Miss. Thamine	30.0000	1	0	2665	14.4542	NaN	
1306	3	0	Zakarian, Mr. Mapriededer	26.5000	0	0	2656	7.2250	NaN	
1307	3	0	Zakarian, Mr. Ortin	27.0000	0	0	2670	7.2250	NaN	
1308	3	0	Zimmerman, Mr. Leo	29.0000	0	0	315082	7.8750	NaN	

1309 rows × 15 columns

In [33]:
```
# Creamos una función que haga este proceso

def createDummies(df, var):
    dummy = pd.get_dummies(df[var], prefix=var)
    df = df.drop(var, axis=1)
    df = pd.concat([df, dummy], axis=1)
    return df
```

In [34]:
```
# Lo probamos con data3
createDummies(data3, 'sex')
```

Out[34]:

	pclass	survived	name	age	sibsp	parch	ticket	fare	cabin	emba
0	1	1	Allen, Miss. Elisabeth Walton	29.0000	0	0	24160	211.3375	B5	
1	1	1	Allison, Master. Hudson Trevor	0.9167	1	2	113781	151.5500	C22 C26	
2	1	0	Allison, Miss. Helen Loraine	2.0000	1	2	113781	151.5500	C22 C26	
3	1	0	Allison, Mr. Hudson Joshua Creighton	30.0000	1	2	113781	151.5500	C22 C26	
4	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	25.0000	1	2	113781	151.5500	C22 C26	
5	1	1	Anderson, Mr. Harry	48.0000	0	0	19952	26.5500	E12	
6	1	1	Andrews, Miss. Kornelia Theodosia	63.0000	1	0	13502	77.9583	D7	
7	1	0	Andrews, Mr. Thomas Jr	39.0000	0	0	112050	0.0000	A36	
8	1	1	Appleton, Mrs. Edward Dale (Charlotte Lamson)	53.0000	2	0	11769	51.4792	C101	
9	1	0	Artagaveytia, Mr. Ramon	71.0000	0	0	PC 17609	49.5042	NaN	
10	1	0	Astor, Col. John Jacob	47.0000	1	0	PC 17757	227.5250	C62 C64	
11	1	1	Astor, Mrs. John Jacob (Madeleine Talmadge Force)	18.0000	1	0	PC 17757	227.5250	C62 C64	
12	1	1	Aubart, Mme. Leontine Pauline	24.0000	0	0	PC 17477	69.3000	B35	
13	1	1	Barber, Miss. Ellen "Nellie"	26.0000	0	0	19877	78.8500	NaN	
14	1	1	Barkworth, Mr. Algernon Henry Wilson	80.0000	0	0	27042	30.0000	A23	
15	1	0	Baumann, Mr. John D	30.0000	0	0	PC 17318	25.9250	NaN	

	pclass	survived	name	age	sibsp	parch	ticket	fare	cabin	emba
16	1	0	Baxter, Mr. Quigg Edmond	24.0000	0	1	PC 17558	247.5208	B58 B60	
17	1	1	Baxter, Mrs. James (Helene DeLaudeniere Chaput)	50.0000	0	1	PC 17558	247.5208	B58 B60	
18	1	1	Bazzani, Miss. Albina	32.0000	0	0	11813	76.2917	D15	
19	1	0	Beattie, Mr. Thomson	36.0000	0	0	13050	75.2417	C6	
20	1	1	Beckwith, Mr. Richard Leonard	37.0000	1	1	11751	52.5542	D35	
21	1	1	Beckwith, Mrs. Richard Leonard (Sallie Monypeny)	47.0000	1	1	11751	52.5542	D35	
22	1	1	Behr, Mr. Karl Howell	26.0000	0	0	111369	30.0000	C148	
23	1	1	Bidois, Miss. Rosalie	42.0000	0	0	PC 17757	227.5250	NaN	
24	1	1	Bird, Miss. Ellen	29.0000	0	0	PC 17483	221.7792	C97	
25	1	0	Birnbaum, Mr. Jakob	25.0000	0	0	13905	26.0000	NaN	
26	1	1	Bishop, Mr. Dickinson H	25.0000	1	0	11967	91.0792	B49	
27	1	1	Bishop, Mrs. Dickinson H (Helen Walton)	19.0000	1	0	11967	91.0792	B49	
28	1	1	Bissette, Miss. Amelia	35.0000	0	0	PC 17760	135.6333	C99	
29	1	1	Bjornstrom-Steffansson, Mr. Mauritz Hakan	28.0000	0	0	110564	26.5500	C52	
...	...	...	...	...	...	...	...	...	...	
1279	3	0	Vestrom, Miss. Hulda Amanda Adolfina	14.0000	0	0	350406	7.8542	NaN	
1280	3	0	Vovk, Mr. Janko	22.0000	0	0	349252	7.8958	NaN	
1281	3	0	Waelens, Mr. Achille	22.0000	0	0	345767	9.0000	NaN	

	pclass	survived	name	age	sibsp	parch	ticket	fare	cabin	emba
1282	3	0	Ware, Mr. Frederick	30.0000	0	0	359309	8.0500	NaN	
1283	3	0	Warren, Mr. Charles William	30.0000	0	0	C.A. 49867	7.5500	NaN	
1284	3	0	Webber, Mr. James	30.0000	0	0	SOTON/OQ 3101316	8.0500	NaN	
1285	3	0	Wenzel, Mr. Linhart	32.5000	0	0	345775	9.5000	NaN	
1286	3	1	Whabee, Mrs. George Joseph (Shawneene Abi-Saab)	38.0000	0	0	2688	7.2292	NaN	
1287	3	0	Widegren, Mr. Carl/Charles Peter	51.0000	0	0	347064	7.7500	NaN	
1288	3	0	Wiklund, Mr. Jakob Alfred	18.0000	1	0	3101267	6.4958	NaN	
1289	3	0	Wiklund, Mr. Karl Johan	21.0000	1	0	3101266	6.4958	NaN	
1290	3	1	Wilkes, Mrs. James (Ellen Needs)	47.0000	1	0	363272	7.0000	NaN	
1291	3	0	Willer, Mr. Aaron ("Abi Weller")	30.0000	0	0	3410	8.7125	NaN	
1292	3	0	Willey, Mr. Edward	30.0000	0	0	S.O./P.P. 751	7.5500	NaN	
1293	3	0	Williams, Mr. Howard Hugh "Harry"	30.0000	0	0	A/5 2466	8.0500	NaN	
1294	3	0	Williams, Mr. Leslie	28.5000	0	0	54636	16.1000	NaN	
1295	3	0	Windelov, Mr. Einar	21.0000	0	0	SOTON/OQ 3101317	7.2500	NaN	
1296	3	0	Wirz, Mr. Albert	27.0000	0	0	315154	8.6625	NaN	
1297	3	0	Wiseman, Mr. Phillippe	30.0000	0	0	A/4. 34244	7.2500	NaN	
1298	3	0	Wittevrongel, Mr. Camille	36.0000	0	0	345771	9.5000	NaN	
1299	3	0	Yasbeck, Mr. Antoni	27.0000	1	0	2659	14.4542	NaN	
1300	3	1	Yasbeck, Mrs. Antoni (Selini Alexander)	15.0000	1	0	2659	14.4542	NaN	
1301	3	0	Youseff, Mr. Gerious	45.5000	0	0	2628	7.2250	NaN	

	pclass	survived	name	age	sibsp	parch	ticket	fare	cabin	emba
1302	3	0	Yousif, Mr. Wazli	30.0000	0	0	2647	7.2250	NaN	
1303	3	0	Yousseff, Mr. Gerious	30.0000	0	0	2627	14.4583	NaN	
1304	3	0	Zabour, Miss. Hileni	14.5000	1	0	2665	14.4542	NaN	
1305	3	0	Zabour, Miss. Thamine	30.0000	1	0	2665	14.4542	NaN	
1306	3	0	Zakarian, Mr. Mapriededer	26.5000	0	0	2656	7.2250	NaN	
1307	3	0	Zakarian, Mr. Ortin	27.0000	0	0	2670	7.2250	NaN	
1308	3	0	Zimmerman, Mr. Leo	29.0000	0	0	315082	7.8750	NaN	

1309 rows × 15 columns

Primeros gráficos

In [35]:
```
df_churn = pd.read_csv("/Users/fsanmartin/python-ml-course-master/datasets/customer-churn-model/Customer Churn Model.txt")
```

In [36]:
```
df_churn.head()
```

Out[36]:

	State	Account Length	Area Code	Phone	Int'l Plan	VMail Plan	VMail Message	Day Mins	Day Calls	Day Charge	...	Eve Calls	Eve Charge
0	KS	128	415	382-4657	no	yes	25	265.1	110	45.07	...	99	16.78
1	OH	107	415	371-7191	no	yes	26	161.6	123	27.47	...	103	16.62
2	NJ	137	415	358-1921	no	no	0	243.4	114	41.38	...	110	10.30
3	OH	84	408	375-9999	yes	no	0	299.4	71	50.90	...	88	5.26
4	OK	75	415	330-6626	yes	no	0	166.7	113	28.34	...	122	12.61

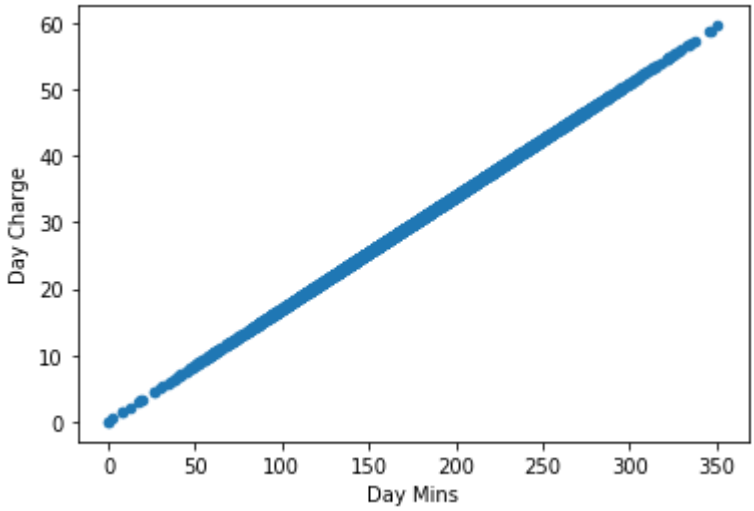
5 rows × 21 columns

In [37]:
```
import matplotlib.pyplot as plt
```

Scatter Plot

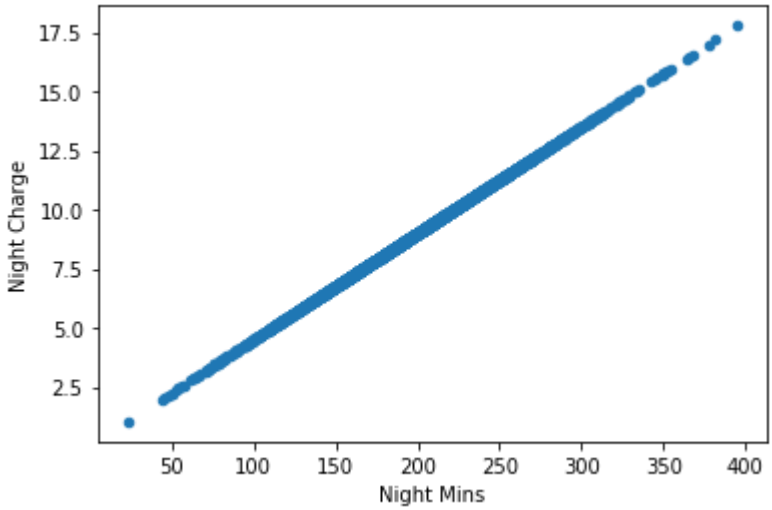
In [38]:
```
df_churn.plot(kind='scatter', x='Day Mins', y='Day Charge')
```

Out[38]: <matplotlib.axes._subplots.AxesSubplot at 0x2163b2ec5c0>



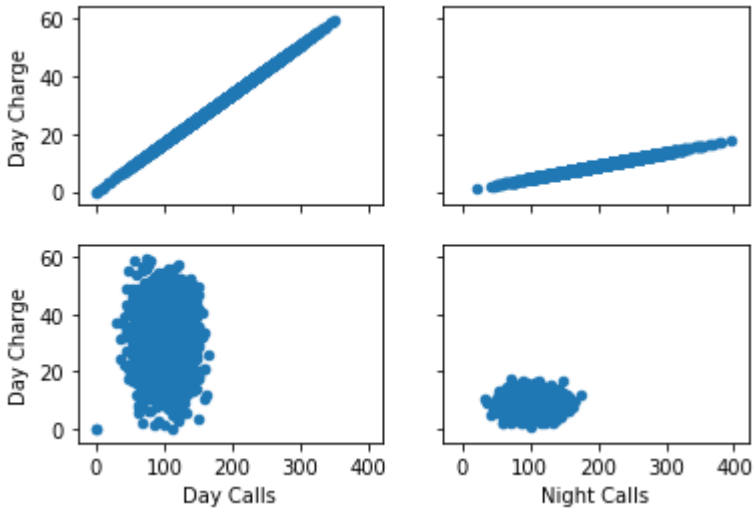
In [39]:
```
df_churn.plot(kind='scatter', x='Night Mins', y='Night Charge')
```

Out[39]: <matplotlib.axes._subplots.AxesSubplot at 0x2163b5dbb00>



In [40]:
```
figure, axs = plt.subplots(2, 2, sharey=True, sharex=True)
df_churn.plot(kind='scatter', x='Day Mins', y='Day Charge', ax=axs[0][0])
df_churn.plot(kind='scatter', x='Night Mins', y='Night Charge', ax=axs[0][1])
df_churn.plot(kind='scatter', x='Day Calls', y='Day Charge', ax=axs[1][0])
df_churn.plot(kind='scatter', x='Night Calls', y='Night Charge', ax=axs[1][1])
```

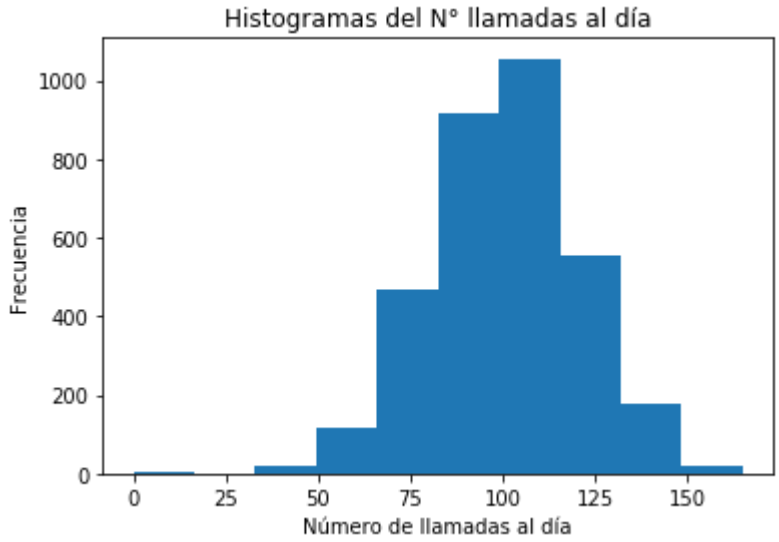
Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x2163c708240>



Histogramas

In [41]:
```
plt.hist(df_churn['Day Calls'])
plt.xlabel('Número de llamadas al día')
plt.ylabel('Frecuencia')
plt.title('Histogramas del N° llamadas al día')
```

Out[41]: Text(0.5, 1.0, 'Histogramas del N° llamadas al día')



¿Cuántos bins se recomienda dejar? Regla de Sturges es la respuesta.

$$c = 1 + \log_2(M)$$

Dónde;

- c = Número de bins
- M = Tamaño de la muestra

In [42]:
```
import numpy as np
```

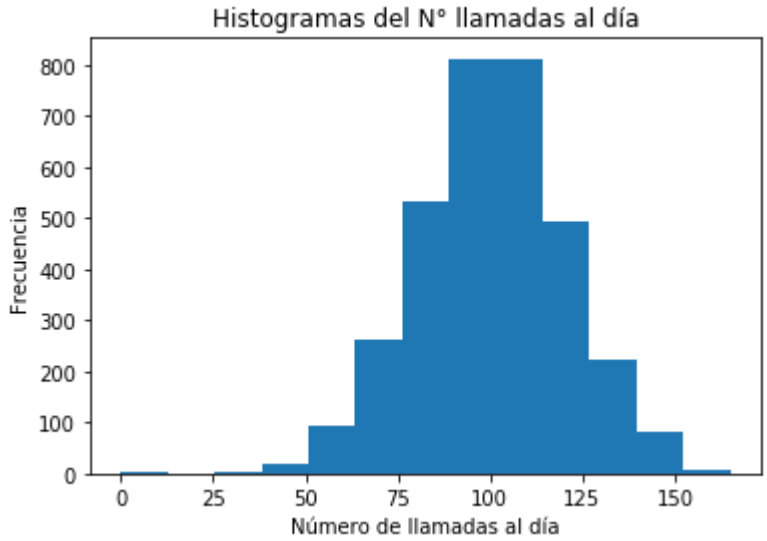
In [43]:
```
c_bins = int(round(1 + np.log2(len(df_churn)), 0))
c_bins
```

Out[43]: 13

Volvemos a graficas el histogramas siguiendo la regla de Sturges

In [44]:
```
plt.hist(df_churn['Day Calls'], bins= c_bins)
plt.xlabel('Número de llamadas al día')
plt.ylabel('Frecuencia')
plt.title('Histogramas del N° llamadas al día')
```

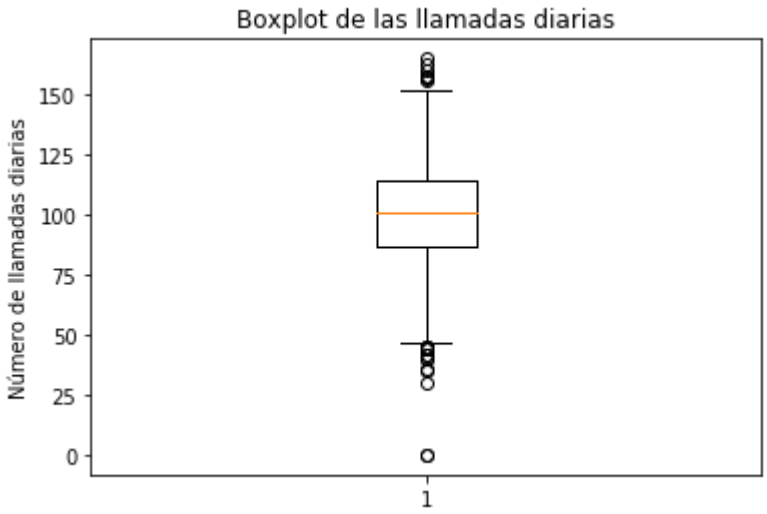
Out[44]: Text(0.5, 1.0, 'Histogramas del N° llamadas al día')



Boxplot

In [45]:
```
plt.boxplot(df_churn['Day Calls'])
plt.ylabel('Número de llamadas diarias')
plt.title('Boxplot de las llamadas diarias')
```

Out[45]: Text(0.5, 1.0, 'Boxplot de las llamadas diarias')



¿Cómo se interpreta?

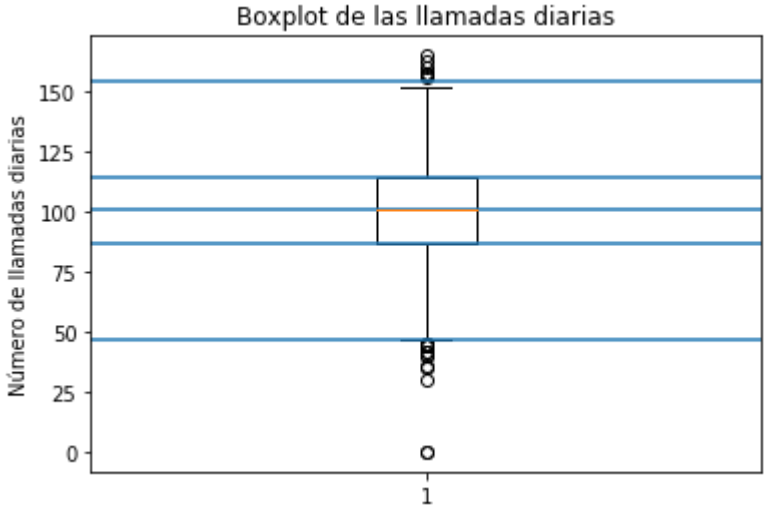
In [46]:
```
df_churn['Day Calls'].describe()
```

Out[46]:
```
count    3333.000000
mean      100.435644
std        20.069084
min         0.000000
25%        87.000000
50%       101.000000
75%       114.000000
max       165.000000
Name: Day Calls, dtype: float64
```

In [47]:
```
#Rango intercuatílico
IQR = df_churn['Day Calls'].quantile(0.75) - df_churn['Day Calls'].quantile(0.
25)

#Valores del boxplot
Lim_inf = df_churn['Day Calls'].quantile(0.25) - 1.5*IQR
Lim_sup = df_churn['Day Calls'].quantile(0.75) + 1.5*IQR
quart_25 = df_churn['Day Calls'].quantile(0.25)
quart_50 = df_churn['Day Calls'].quantile(0.50)
quart_75 = df_churn['Day Calls'].quantile(0.75)
valores_box = [Lim_inf, Lim_sup, quart_25, quart_50, quart_75]
```

In [48]:
```
plt.boxplot(df_churn['Day Calls'])
plt.ylabel('Número de llamadas diarias')
plt.title('Boxplot de las llamadas diarias')
for valor in valores_box:
    plt.axhline(valor)
```



Cualquier valor que esté bajo 'Lim_inf' o sobre 'Lim_sup' se considera un "Outlier"

In [ ]:

In [ ]: