In [4]:

```
import pandas as pd
# Note that pd.read_csv is used because we imported pandas as pd
pd.read_csv(″surveys.csv″)
```

Out[4]:

	record_id	month	day	year	plot_id	species_id	sex	hindfoot_length	weight
0	1	7	16	1977	2	NL	M	32.0	NaN
1	2	7	16	1977	3	NL	M	33.0	NaN
2	3	7	16	1977	2	DM	F	37.0	NaN
3	4	7	16	1977	7	DM	M	36.0	NaN
4	5	7	16	1977	3	DM	M	35.0	NaN
5	6	7	16	1977	1	PF	M	14.0	NaN
6	7	7	16	1977	2	PE	F	NaN	NaN
7	8	7	16	1977	1	DM	M	37.0	NaN
8	9	7	16	1977	1	DM	F	34.0	NaN
9	10	7	16	1977	6	PF	F	20.0	NaN
10	11	7	16	1977	5	DS	F	53.0	NaN
11	12	7	16	1977	7	DM	M	38.0	NaN
12	13	7	16	1977	3	DM	M	35.0	NaN
13	14	7	16	1977	8	DM	NaN	NaN	NaN
14	15	7	16	1977	6	DM	F	36.0	NaN
15	16	7	16	1977	4	DM	F	36.0	NaN
16	17	7	16	1977	3	DS	F	48.0	NaN
17	18	7	16	1977	2	PP	M	22.0	NaN
18	19	7	16	1977	4	PF	NaN	NaN	NaN
19	20	7	17	1977	11	DS	F	48.0	NaN
20	21	7	17	1977	14	DM	F	34.0	NaN
21	22	7	17	1977	15	NL	F	31.0	NaN
22	23	7	17	1977	13	DM	M	36.0	NaN
23	24	7	17	1977	13	SH	M	21.0	NaN
24	25	7	17	1977	9	DM	M	35.0	NaN
25	26	7	17	1977	15	DM	M	31.0	NaN
26	27	7	17	1977	15	DM	M	36.0	NaN
27	28	7	17	1977	11	DM	M	38.0	NaN
28	29	7	17	1977	11	PP	M	NaN	NaN
29	30	7	17	1977	10	DS	F	52.0	NaN
...	...	...	...	...	...	...	...	...	...
35519	35520	12	31	2002	9	SF	NaN	24.0	36.0
35520	35521	12	31	2002	9	DM	M	37.0	48.0
35521	35522	12	31	2002	9	DM	F	35.0	45.0
35522	35523	12	31	2002	9	DM	F	36.0	44.0
35523	35524	12	31	2002	9	PB	F	25.0	27.0
35524	35525	12	31	2002	9	OL	M	21.0	26.0
35525	35526	12	31	2002	8	OT	F	20.0	24.0

	record_id	month	day	year	plot_id	species_id	sex	hindfoot_length	weight
35526	35527	12	31	2002	13	DO	F	33.0	43.0
35527	35528	12	31	2002	13	US	NaN	NaN	NaN
35528	35529	12	31	2002	13	PB	F	25.0	25.0
35529	35530	12	31	2002	13	OT	F	20.0	NaN
35530	35531	12	31	2002	13	PB	F	27.0	NaN
35531	35532	12	31	2002	14	DM	F	34.0	43.0
35532	35533	12	31	2002	14	DM	F	36.0	48.0
35533	35534	12	31	2002	14	DM	M	37.0	56.0
35534	35535	12	31	2002	14	DM	M	37.0	53.0
35535	35536	12	31	2002	14	DM	F	35.0	42.0
35536	35537	12	31	2002	14	DM	F	36.0	46.0
35537	35538	12	31	2002	15	PB	F	26.0	31.0
35538	35539	12	31	2002	15	SF	M	26.0	68.0
35539	35540	12	31	2002	15	PB	F	26.0	23.0
35540	35541	12	31	2002	15	PB	F	24.0	31.0
35541	35542	12	31	2002	15	PB	F	26.0	29.0
35542	35543	12	31	2002	15	PB	F	27.0	34.0
35543	35544	12	31	2002	15	US	NaN	NaN	NaN
35544	35545	12	31	2002	15	AH	NaN	NaN	NaN
35545	35546	12	31	2002	15	AH	NaN	NaN	NaN
35546	35547	12	31	2002	10	RM	F	15.0	14.0
35547	35548	12	31	2002	7	DO	M	36.0	51.0
35548	35549	12	31	2002	5	NaN	NaN	NaN	NaN

35549 rows × 9 columns

```
In [6]: surveys_df = pd.read_csv("surveys.csv")
```

In [7]:

```
surveys_df
```

Out[7]:

	record_id	month	day	year	plot_id	species_id	sex	hindfoot_length	weight
0	1	7	16	1977	2	NL	M	32.0	NaN
1	2	7	16	1977	3	NL	M	33.0	NaN
2	3	7	16	1977	2	DM	F	37.0	NaN
3	4	7	16	1977	7	DM	M	36.0	NaN
4	5	7	16	1977	3	DM	M	35.0	NaN
5	6	7	16	1977	1	PF	M	14.0	NaN
6	7	7	16	1977	2	PE	F	NaN	NaN
7	8	7	16	1977	1	DM	M	37.0	NaN
8	9	7	16	1977	1	DM	F	34.0	NaN
9	10	7	16	1977	6	PF	F	20.0	NaN
10	11	7	16	1977	5	DS	F	53.0	NaN
11	12	7	16	1977	7	DM	M	38.0	NaN
12	13	7	16	1977	3	DM	M	35.0	NaN
13	14	7	16	1977	8	DM	NaN	NaN	NaN
14	15	7	16	1977	6	DM	F	36.0	NaN
15	16	7	16	1977	4	DM	F	36.0	NaN
16	17	7	16	1977	3	DS	F	48.0	NaN
17	18	7	16	1977	2	PP	M	22.0	NaN
18	19	7	16	1977	4	PF	NaN	NaN	NaN
19	20	7	17	1977	11	DS	F	48.0	NaN
20	21	7	17	1977	14	DM	F	34.0	NaN
21	22	7	17	1977	15	NL	F	31.0	NaN
22	23	7	17	1977	13	DM	M	36.0	NaN
23	24	7	17	1977	13	SH	M	21.0	NaN
24	25	7	17	1977	9	DM	M	35.0	NaN
25	26	7	17	1977	15	DM	M	31.0	NaN
26	27	7	17	1977	15	DM	M	36.0	NaN
27	28	7	17	1977	11	DM	M	38.0	NaN
28	29	7	17	1977	11	PP	M	NaN	NaN
29	30	7	17	1977	10	DS	F	52.0	NaN
...	...	...	...	...	...	...	...	...	...
35519	35520	12	31	2002	9	SF	NaN	24.0	36.0
35520	35521	12	31	2002	9	DM	M	37.0	48.0
35521	35522	12	31	2002	9	DM	F	35.0	45.0
35522	35523	12	31	2002	9	DM	F	36.0	44.0
35523	35524	12	31	2002	9	PB	F	25.0	27.0
35524	35525	12	31	2002	9	OL	M	21.0	26.0
35525	35526	12	31	2002	8	OT	F	20.0	24.0

	record_id	month	day	year	plot_id	species_id	sex	hindfoot_length	weight
35526	35527	12	31	2002	13	DO	F	33.0	43.0
35527	35528	12	31	2002	13	US	NaN	NaN	NaN
35528	35529	12	31	2002	13	PB	F	25.0	25.0
35529	35530	12	31	2002	13	OT	F	20.0	NaN
35530	35531	12	31	2002	13	PB	F	27.0	NaN
35531	35532	12	31	2002	14	DM	F	34.0	43.0
35532	35533	12	31	2002	14	DM	F	36.0	48.0
35533	35534	12	31	2002	14	DM	M	37.0	56.0
35534	35535	12	31	2002	14	DM	M	37.0	53.0
35535	35536	12	31	2002	14	DM	F	35.0	42.0
35536	35537	12	31	2002	14	DM	F	36.0	46.0
35537	35538	12	31	2002	15	PB	F	26.0	31.0
35538	35539	12	31	2002	15	SF	M	26.0	68.0
35539	35540	12	31	2002	15	PB	F	26.0	23.0
35540	35541	12	31	2002	15	PB	F	24.0	31.0
35541	35542	12	31	2002	15	PB	F	26.0	29.0
35542	35543	12	31	2002	15	PB	F	27.0	34.0
35543	35544	12	31	2002	15	US	NaN	NaN	NaN
35544	35545	12	31	2002	15	AH	NaN	NaN	NaN
35545	35546	12	31	2002	15	AH	NaN	NaN	NaN
35546	35547	12	31	2002	10	RM	F	15.0	14.0
35547	35548	12	31	2002	7	DO	M	36.0	51.0
35548	35549	12	31	2002	5	NaN	NaN	NaN	NaN

35549 rows × 9 columns

In [8]:

```
surveys_df.head()
```

Out[8]:

	record_id	month	day	year	plot_id	species_id	sex	hindfoot_length	weight
0	1	7	16	1977	2	NL	M	32.0	NaN
1	2	7	16	1977	3	NL	M	33.0	NaN
2	3	7	16	1977	2	DM	F	37.0	NaN
3	4	7	16	1977	7	DM	M	36.0	NaN
4	5	7	16	1977	3	DM	M	35.0	NaN

In [9]:

```
type(surveys_df)
```

Out[9]:

```
pandas.core.frame.DataFrame
```

```
In [10]: surveys_df.dtypes
```

```
Out[10]: record_id            int64
         month                int64
         day                  int64
         year                 int64
         plot_id              int64
         species_id          object
         sex                 object
         hindfoot_length    float64
         weight             float64
         dtype: object
```

```
In [11]: surveys_df.columns
```

```
Out[11]: Index(['record_id', 'month', 'day', 'year', 'plot_id', 'species_id', 'sex',
                'hindfoot_length', 'weight'],
               dtype='object')
```

```
In [12]: surveys_df.shape
```

```
Out[12]: (35549, 9)
```

```
In [13]: surveys_df.tail()
```

Out[13]:

	record_id	month	day	year	plot_id	species_id	sex	hindfoot_length	weight
35544	35545	12	31	2002	15	AH	NaN	NaN	NaN
35545	35546	12	31	2002	15	AH	NaN	NaN	NaN
35546	35547	12	31	2002	10	RM	F	15.0	14.0
35547	35548	12	31	2002	7	DO	M	36.0	51.0
35548	35549	12	31	2002	5	NaN	NaN	NaN	NaN

```
In [14]: # Look at the column names
         surveys_df.columns
```

```
Out[14]: Index(['record_id', 'month', 'day', 'year', 'plot_id', 'species_id', 'sex',
                'hindfoot_length', 'weight'],
               dtype='object')
```

```
In [15]: pd.unique(surveys_df['species_id'])
```

```
Out[15]: array(['NL', 'DM', 'PF', 'PE', 'DS', 'PP', 'SH', 'OT', 'DO', 'OX', 'SS',
                'OL', 'RM', nan, 'SA', 'PM', 'AH', 'DX', 'AB', 'CB', 'CM', 'CQ',
                'RF', 'PC', 'PG', 'PH', 'PU', 'CV', 'UR', 'UP', 'ZL', 'UL', 'CS',
                'SC', 'BA', 'SF', 'RO', 'AS', 'SO', 'PI', 'ST', 'CU', 'SU', 'RX',
                'PB', 'PL', 'PX', 'CT', 'US'], dtype=object)
```

In [16]:
```
surveys_df['weight'].describe()
```

Out[16]:
```
count    32283.000000
mean        42.672428
std         36.631259
min          4.000000
25%         20.000000
50%         37.000000
75%         48.000000
max        280.000000
Name: weight, dtype: float64
```

In [18]:
```
surveys_df['weight'].min()
```

Out[18]:
```
4.0
```

In [19]:
```
surveys_df['weight'].max()
```

Out[19]:
```
280.0
```

In [20]:
```
surveys_df['weight'].mean()
```

Out[20]:
```
42.672428212991356
```

In [21]:
```
surveys_df['weight'].std()
```

Out[21]:
```
36.63125947458399
```

In [22]:
```
surveys_df['weight'].count()
```

Out[22]:
```
32283
```

In [23]:
```
# Group data by sex
grouped_data = surveys_df.groupby('sex')
```

In [24]:
```
grouped_data.describe()
```

Out[24]:

	day								hindfoot_length		...	weight	
	count	mean	std	min	25%	50%	75%	max	count	mean	...	75%	ma
sex													
F	15690.0	16.007138	8.271144	1.0	9.0	16.0	23.0	31.0	14894.0	28.836780	...	46.0	27
M	17348.0	16.184286	8.199274	1.0	9.0	16.0	23.0	31.0	16476.0	29.709578	...	49.0	28

2 rows × 56 columns

In [25]:
```
grouped_data.mean(numeric_only=True)
```

Out[25]:

	record_id	month	day	year	plot_id	hindfoot_length	weight
sex							
F	18036.412046	6.583047	16.007138	1990.644997	11.440854	28.836780	42.170555
M	17754.835601	6.392668	16.184286	1990.480401	11.098282	29.709578	42.995379

In [26]:
```
grouped_data2 = surveys_df.groupby(['plot_id', 'sex'])
```

In [27]:

```
grouped_data2.mean(numeric_only=True)
```

Out[27]:

		record_id	month	day	year	hindfoot_length	weight
plot_id	**sex**						
1	**F**	18390.384434	6.597877	15.338443	1990.933962	31.733911	46.311138
	M	17197.740639	6.121461	15.905936	1990.091324	34.302770	55.950560
2	**F**	17714.753608	6.426804	16.288660	1990.449485	30.161220	52.561845
	M	18085.458042	6.340035	15.440559	1990.756119	30.353760	51.391382
3	**F**	19888.783875	6.604703	16.161254	1992.013438	23.774044	31.215349
	M	20226.767857	6.271429	16.450000	1992.275000	23.833744	34.163241
4	**F**	17489.205275	6.442661	15.746560	1990.235092	33.249102	46.818824
	M	18493.841748	6.430097	16.507767	1991.000971	34.097959	48.888119
5	**F**	12280.793169	6.142315	15.722960	1986.485769	28.921844	40.974806
	M	12798.426621	6.194539	15.703072	1986.817406	29.694794	40.708551
6	**F**	19406.503392	6.628223	16.313433	1991.579376	26.981322	36.352288
	M	17849.574607	6.035340	16.091623	1990.556283	27.425591	36.867388
7	**F**	19069.668657	6.385075	15.313433	1991.441791	19.779553	20.006135
	M	19188.729642	6.719870	15.778502	1991.462541	20.536667	21.194719
8	**F**	18920.276190	6.632143	15.836905	1991.267857	32.187578	45.623011
	M	19452.109868	6.571719	15.854527	1991.686673	33.751059	49.641372
9	**F**	16217.497069	6.499414	15.555686	1989.303634	35.126092	53.618469
	M	18000.710159	6.361554	15.209163	1990.632470	34.175732	49.519309
10	**F**	16001.496454	5.588652	16.964539	1989.248227	18.641791	17.094203
	M	15708.704225	5.718310	16.739437	1989.007042	19.567164	19.971223
11	**F**	16994.962287	6.759124	16.283455	1989.836983	32.029299	43.515075
	M	16933.909621	6.374150	15.974733	1989.856171	32.078014	43.366197
12	**F**	17457.966981	6.509434	16.305660	1990.266981	30.975124	49.831731
	M	17592.327500	6.304167	16.367500	1990.400833	31.762489	48.909710
13	**F**	18033.100318	6.802548	16.229299	1990.619427	27.201014	40.524590
	M	16969.044700	6.480204	16.005109	1989.911877	27.893793	40.097754
14	**F**	17097.145275	6.510578	16.681241	1989.974612	32.973373	47.355491
	M	17891.948598	6.660748	16.504673	1990.587850	32.961802	45.159378
15	**F**	20602.449064	6.569647	16.162162	1992.523909	21.949891	26.670236
	M	18104.019560	6.185819	17.413203	1990.770171	21.803109	27.523691
16	**F**	19002.445946	6.360360	16.819820	1991.351351	23.144928	25.810427
	M	18434.714286	6.201465	16.622711	1990.926740	23.480916	23.811321
17	**F**	18234.322870	6.650224	15.892377	1990.785874	30.918536	48.176201
	M	18857.651472	6.569801	16.183286	1991.331434	32.227634	47.558853
18	**F**	17940.875497	6.698013	15.960265	1990.536424	26.690341	36.963514
	M	15106.718850	6.610224	16.797125	1988.551118	27.703072	43.546952

		record_id	month	day	year	hindfoot_length	weight
plot_id	**sex**						
19	**F**	21848.216475	6.701149	15.226054	1993.417625	21.257937	21.978599
	M	19470.779690	6.533563	16.647160	1991.740103	21.071685	20.306878
20	**F**	17510.769231	6.743077	16.026154	1990.253846	27.069193	52.624406
	M	16076.192496	6.489396	16.375204	1989.243067	27.908451	44.197279
21	**F**	22452.636661	6.860884	16.307692	1993.878887	22.366554	25.974832
	M	20120.399113	6.671840	16.203991	1992.199557	21.736721	22.772622
22	**F**	18499.695976	6.651267	15.521610	1990.973174	34.108320	53.647059
	M	18015.365527	6.381872	16.682021	1990.650817	33.359746	54.572531
23	**F**	15863.193939	6.860606	16.036364	1989.024242	20.051948	20.564417
	M	17091.338164	6.391304	16.077295	1989.961353	19.850000	18.941463
24	**F**	13702.224280	6.596708	16.393004	1987.485597	26.993377	47.914405
	M	15208.136082	6.360825	16.971134	1988.641237	25.786996	39.321503

In [28]:
```
# Count the number of samples by species
species_counts = surveys_df.groupby('species_id')['record_id'].count()
print(species_counts)
```

```
species_id
AB      303
AH      437
AS        2
BA       46
CB       50
CM       13
CQ       16
CS        1
CT        1
CU        1
CV        1
DM    10596
DO     3027
DS     2504
DX       40
NL     1252
OL     1006
OT     2249
OX       12
PB     2891
PC       39
PE     1299
PF     1597
PG        8
PH       32
PI        9
PL       36
PM      899
PP     3123
PU        5
PX        6
RF       75
RM     2609
RO        8
RX        2
SA       75
SC        1
SF       43
SH      147
SO       43
SS      248
ST        1
SU        5
UL        4
UP        8
UR       10
US        4
ZL        2
Name: record_id, dtype: int64
```

In [29]:
```
surveys_df.groupby('species_id')['record_id'].count()['DO']
```

Out[29]:
```
3027
```

In [30]:

```
# Multiply all weight values by 2
surveys_df['weight']*2
```

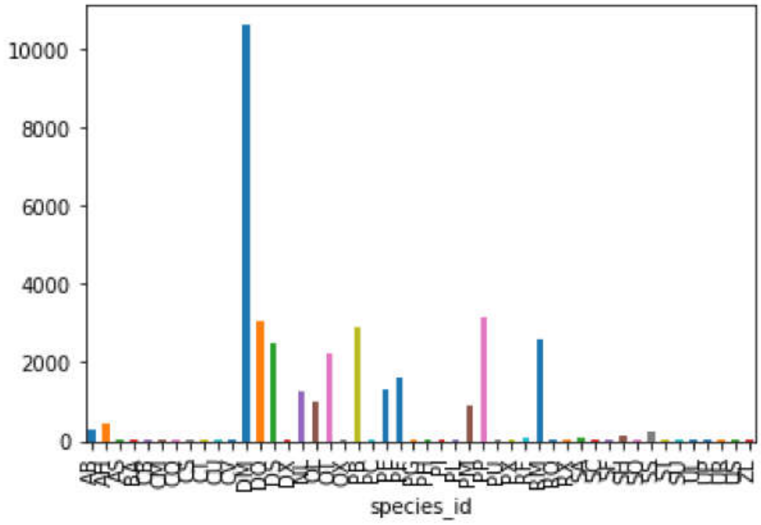
Out[30]:

```
0          NaN
1          NaN
2          NaN
3          NaN
4          NaN
5          NaN
6          NaN
7          NaN
8          NaN
9          NaN
10         NaN
11         NaN
12         NaN
13         NaN
14         NaN
15         NaN
16         NaN
17         NaN
18         NaN
19         NaN
20         NaN
21         NaN
22         NaN
23         NaN
24         NaN
25         NaN
26         NaN
27         NaN
28         NaN
29         NaN
         ...  
35519     72.0
35520     96.0
35521     90.0
35522     88.0
35523     54.0
35524     52.0
35525     48.0
35526     86.0
35527      NaN
35528     50.0
35529      NaN
35530      NaN
35531     86.0
35532     96.0
35533    112.0
35534    106.0
35535     84.0
35536     92.0
35537     62.0
35538    136.0
35539     46.0
35540     62.0
35541     58.0
35542     68.0
35543      NaN
35544      NaN
35545      NaN
35546     28.0
35547    102.0
```

```
35548       NaN
Name: weight, Length: 35549, dtype: float64
```

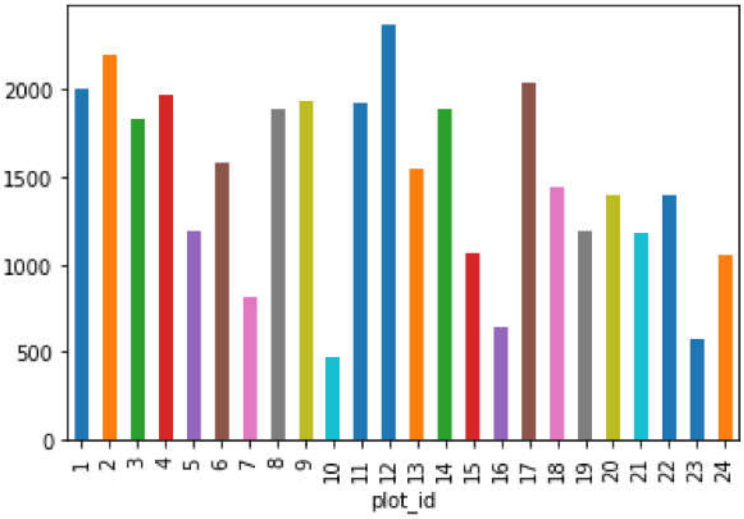
In [31]:

```
# Make sure figures appear inline in Ipython Notebook
%matplotlib inline
# Create a quick bar chart
species_counts.plot(kind='bar');
```



In [32]:

```
total_count = surveys_df.groupby('plot_id')['record_id'].nunique()
# Let's plot that too
total_count.plot(kind='bar');
```



In [33]:

```
d = {'one' : pd.Series([1., 2., 3.], index=['a', 'b', 'c']), 'two' : pd.Series([1., 2.,
pd.DataFrame(d)
```

Out[33]:

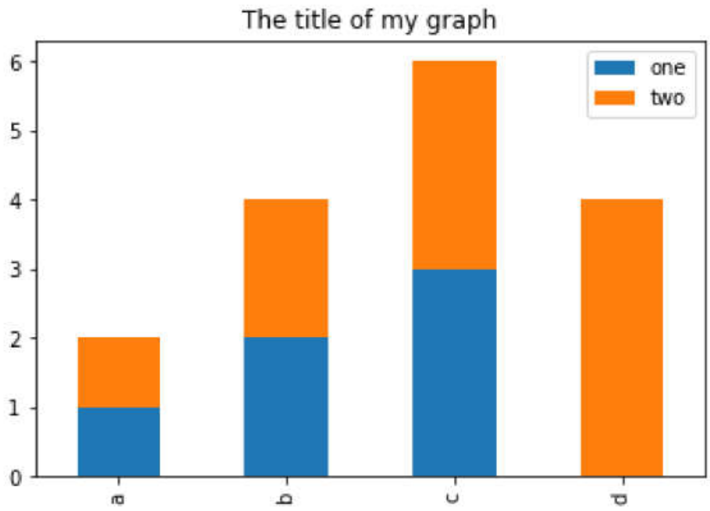
	one	two
a	1.0	1.0
b	2.0	2.0
c	3.0	3.0
d	NaN	4.0

In [34]:

```
# Plot stacked data so columns 'one' and 'two' are stacked
my_df = pd.DataFrame(d)
my_df.plot(kind='bar', stacked=True, title="The title of my graph")
```

Out[34]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ea2c9969e8>
```



In [ ]:

```

```