Ques 14 : How do I explore a pandas Series?

In [1]:
```
import pandas as pd
movie_ratings_ds = pd.read_csv('http://bit.ly/imdbratings')
movie_ratings_ds.head(4)
```

Out[1]:

	star_rating	title	content_rating	genre	duration	actors_list
0	9.3	The Shawshank Redemption	R	Crime	142	[u'Tim Robbins', u'Morgan Freeman', u'Bob Gunt...
1	9.2	The Godfather	R	Crime	175	[u'Marlon Brando', u'Al Pacino', u'James Caan']
2	9.1	The Godfather: Part II	R	Crime	200	[u'Al Pacino', u'Robert De Niro', u'Robert Duv...
3	9.0	The Dark Knight	PG-13	Action	152	[u'Christian Bale', u'Heath Ledger', u'Aaron E...

In [2]:
```
movie_ratings_ds.genre.describe()
```

Out[2]:
```
count       979
unique       16
top       Drama
freq        278
Name: genre, dtype: object
```

In [3]:
```
# showing the count for every distinct value of the 'genre' column
movie_ratings_ds.genre.value_counts()
```

Out[3]:
```
Drama        278
Comedy       156
Action       136
Crime        124
Biography     77
Adventure     75
Animation     62
Horror        29
Mystery       16
Western        9
Thriller       5
Sci-Fi         5
Film-Noir      3
Family         2
History        1
Fantasy        1
Name: genre, dtype: int64
```

In [4]:
```
# showing percentage
movie_ratings_ds.genre.value_counts(normalize=True)
```

Out[4]:
```
Drama        0.283963
Comedy       0.159346
Action       0.138917
Crime        0.126660
Biography    0.078652
Adventure    0.076609
Animation    0.063330
Horror       0.029622
Mystery      0.016343
Western      0.009193
Thriller     0.005107
Sci-Fi       0.005107
Film-Noir    0.003064
Family       0.002043
History      0.001021
Fantasy      0.001021
Name: genre, dtype: float64
```

In [5]:
```
movie_ratings_ds.genre.unique()  # Show all unique values
```

Out[5]:
```
array(['Crime', 'Action', 'Drama', 'Western', 'Adventure', 'Biography',
       'Comedy', 'Animation', 'Mystery', 'Horror', 'Film-Noir', 'Sci-Fi',
       'History', 'Thriller', 'Family', 'Fantasy'], dtype=object)
```

In [6]:
```
movie_ratings_ds.genre.nunique()  # Show the count of unique value
```

Out[6]:
```
16
```

In [8]:
```
# Show the the details content_rating for each genre type
pd.crosstab(movie_ratings_ds.genre, movie_ratings_ds.content_rating)
```

Out[8]:

content_rating	APPROVED	G	GP	NC-17	NOT RATED	PASSED	PG	PG-13	R	TV-MA	UNRATED	X
genre												
Action	3	1	1	0	4	1	11	44	67	0	3	0
Adventure	3	2	0	0	5	1	21	23	17	0	2	0
Animation	3	20	0	0	3	0	25	5	5	0	1	0
Biography	1	2	1	0	1	0	6	29	36	0	0	0
Comedy	9	2	1	1	16	3	23	23	73	0	4	1
Crime	6	0	0	1	7	1	6	4	87	0	11	1
Drama	12	3	0	4	24	1	25	55	143	1	9	1
Family	0	1	0	0	0	0	1	0	0	0	0	0
Fantasy	0	0	0	0	0	0	0	0	1	0	0	0
Film-Noir	1	0	0	0	1	0	0	0	0	0	1	0
History	0	0	0	0	0	0	0	0	0	0	1	0
Horror	2	0	0	1	1	0	1	2	16	0	5	1
Mystery	4	1	0	0	1	0	1	2	6	0	1	0
Sci-Fi	1	0	0	0	0	0	0	1	3	0	0	0
Thriller	1	0	0	0	0	0	1	0	3	0	0	0
Western	1	0	0	0	2	0	2	1	3	0	0	0

In [9]:
```
movie_ratings_ds.duration.describe()
```

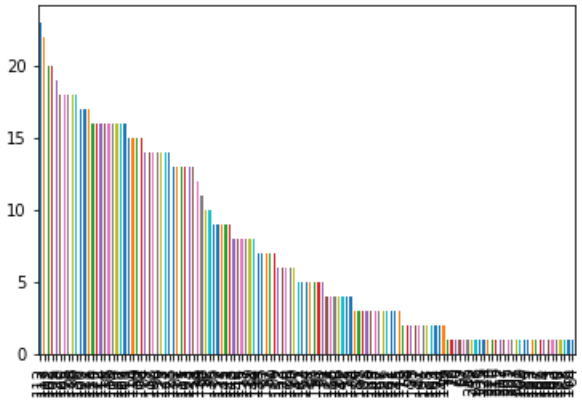
Out[9]:
```
count    979.000000
mean     120.979571
std       26.218010
min       64.000000
25%      102.000000
50%      117.000000
75%      134.000000
max      242.000000
Name: duration, dtype: float64
```

In [10]:
```
movie_ratings_ds.duration.plot(kind='hist')  # A histogram shows the distribution of a numerical variable
```

Out[10]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x49b68fe1d0>
```

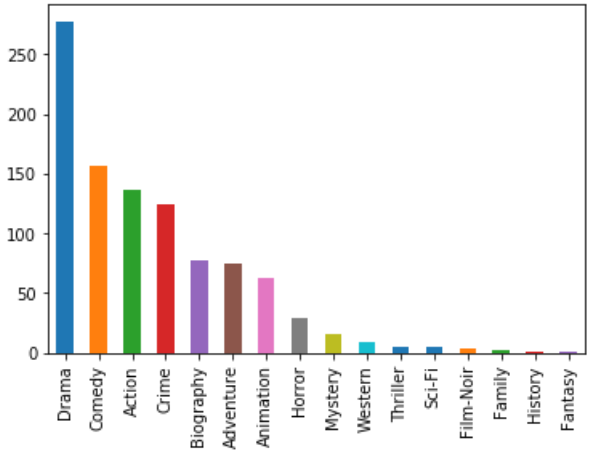
In [11]:
```
# Not feasibale if there is so many distinct value for a column
movie_ratings_ds.duration.value_counts().plot(kind='bar')
```

Out[11]: <matplotlib.axes._subplots.AxesSubplot at 0x49b6a15b38>



In [12]:
```
movie_ratings_ds.genre.value_counts().plot(kind='bar')
```

Out[12]: <matplotlib.axes._subplots.AxesSubplot at 0x49b6dd8160>



In [ ]: