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

```
In [4]: movies = pd.read_csv('https://raw.githubusercontent.com/justmarkham/pandas-vid
        eos/master/data/imdb_1000.csv')
```

```
In [5]: movies.head()
```

Out[5]:

	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...
4	8.9	Pulp Fiction	R	Crime	154	[u'John Travolta', u'Uma Thurman', u'Samuel L....

```
In [6]: #datatypes
        movies.dtypes
```

```
Out[6]: star_rating       float64
        title              object
        content_rating     object
        genre              object
        duration            int64
        actors_list        object
        dtype: object
```

```
In [8]: #Describe  Genre
        movies.genre.describe()
```

```
Out[8]: count       979
        unique       16
        top       Drama
        freq        278
        Name: genre, dtype: object
```

```
In [9]:  #Count the Genre Values
          movies.genre.value_counts()
```

```
Out[9]:  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
         Fantasy        1
         History        1
         Name: genre, dtype: int64
```

```
In [10]:  #Getting the Percentage values of genre counts
          movies.genre.value_counts(normalize=True)
```

```
Out[10]:  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
          Fantasy      0.001021
          History      0.001021
          Name: genre, dtype: float64
```

```
In [11]:  #unique Values in Genre Series
          movies.genre.unique()
```

```
Out[11]:  array(['Crime', 'Action', 'Drama', 'Western', 'Adventure', 'Biography',
                 'Comedy', 'Animation', 'Mystery', 'Horror', 'Film-Noir', 'Sci-Fi',
                 'History', 'Thriller', 'Family', 'Fantasy'], dtype=object)
```

```
In [12]:  #numeric values of Unique Values
          movies.genre.nunique()
```

```
Out[12]:  16
```

In [13]:

```
#CrossTab functions to find the count of results of series of columns Genre An
d PG-Ratings
pd.crosstab(movies.genre, movies.content_rating)
```

Out[13]:

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

In [14]:

```
#describe Duration
movies.duration.describe()
```

Out[14]:

```
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 [15]: #Count values of Duration
         movies.duration.value_counts()
```

```
Out[15]: 112    23
         113    22
         102    20
         101    20
         129    19
         120    18
         105    18
         126    18
         98     18
         130    18
         100    17
         121    17
         116    17
         124    16
         122    16
         118    16
         115    16
         96     16
         104    16
         110    16
         107    16
         109    16
         119    15
         114    15
         99     15
         108    15
         94     14
         117    14
         106    14
         93     14
                ..
         70      1
         69      1
         67      1
         66      1
         242     1
         238     1
         195     1
         229     1
         224     1
         220     1
         216     1
         212     1
         207     1
         205     1
         202     1
         201     1
         200     1
         194     1
         159     1
         193     1
         187     1
         186     1
         184     1
         183     1
         182     1
         180     1
```

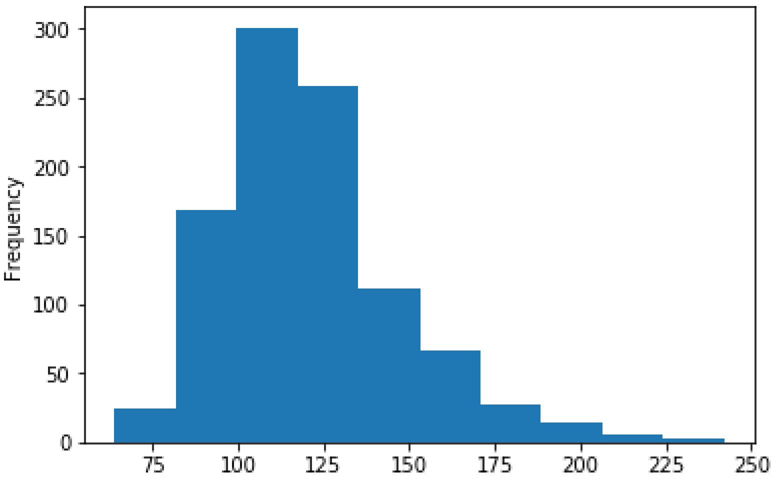
```
177      1
168      1
166      1
64       1
Name: duration, Length: 133, dtype: int64
```

In [17]:

```
#PLot graph using Matplotlib inline
%matplotlib inline
movies.duration.plot(kind = 'hist')
```

Out[17]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1c98f311828>
```

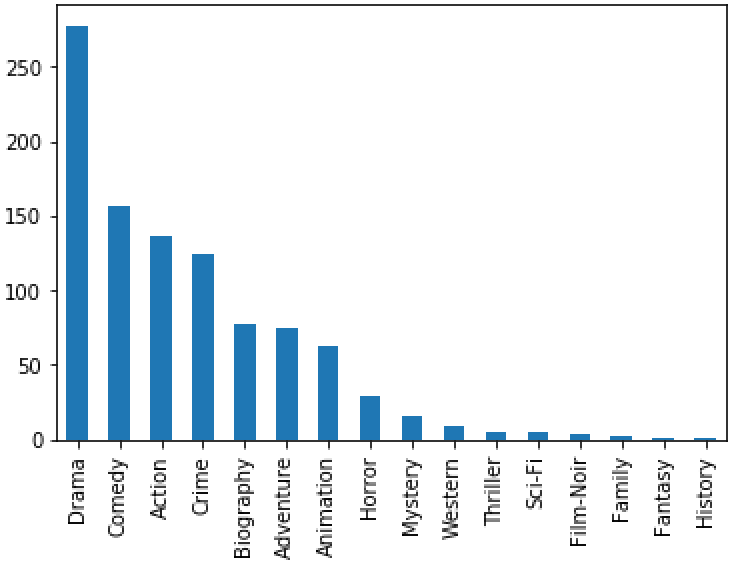


In [18]:

```
#PLot graph for Genre Value Counts
movies.genre.value_counts().plot(kind = 'bar')
```

Out[18]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1c98f3be198>
```


Drama
Comedy
Action
Crime
Biography
Adventure
Animation
Horror
Mystery
Western
Thriller
Sci-Fi
Film-Noir
Family
Fantasy
History