

Для анализа данных возьмем датасет о титанике. Сначала импортируем библиотеку `pandas`, а затем загружаем исходные данные.

Ввод [1]:

```
import pandas as pd
```

Ввод [2]:

```
tit = pd.read_csv('D:/Загрузки/titanic.csv')
```

Проверим, загрузились ли наши данные, распечатав таблицу

Ввод [3]:

```
print(tit)
```

	PassengerID	Name	PClass	\
0	1	Allen, Miss Elisabeth Walton	1st	
1	2	Allison, Miss Helen Loraine	1st	
2	3	Allison, Mr Hudson Joshua Creighton	1st	
3	4	Allison, Mrs Hudson JC (Bessie Waldo Daniels)	1st	
4	5	Allison, Master Hudson Trevor	1st	
...	
1308	1309	Zakarian, Mr Artun	3rd	
1309	1310	Zakarian, Mr Maprieder	3rd	
1310	1311	Zenni, Mr Philip	3rd	
1311	1312	Lievens, Mr Rene	3rd	
1312	1313	Zimmerman, Leo	3rd	

	Age	Sex	Survived	SexCode
0	29.00	female	1	1
1	2.00	female	0	1
2	30.00	male	0	0
3	25.00	female	0	1
4	0.92	male	1	0
...
1308	27.00	male	0	0
1309	26.00	male	0	0
1310	22.00	male	0	0
1311	24.00	male	0	0
1312	29.00	male	0	0

[1313 rows x 7 columns]

Затем воспользуемся функцией `head` что бы распечатать первые элементы таблицы

Ввод [4]:

```
tit.head()
```

Out[4]:

	PassengerID	Name	PClass	Age	Sex	Survived	SexCode
0	1	Allen, Miss Elisabeth Walton	1st	29.00	female	1	1
1	2	Allison, Miss Helen Loraine	1st	2.00	female	0	1
2	3	Allison, Mr Hudson Joshua Creighton	1st	30.00	male	0	0
3	4	Allison, Mrs Hudson JC (Bessie Waldo Daniels)	1st	25.00	female	0	1
4	5	Allison, Master Hudson Trevor	1st	0.92	male	1	0

Что бы узнать размер таблицы, обратимся к функции `shape`

Ввод [5]:

```
tit.shape
```

Out[5]:

```
(1313, 7)
```

Далее повыводим некоторые части таблицы

Ввод [6]:

```
tit['Name']
```

Out[6]:

```
0          Allen, Miss Elisabeth Walton
1          Allison, Miss Helen Loraine
2          Allison, Mr Hudson Joshua Creighton
3  Allison, Mrs Hudson JC (Bessie Waldo Daniels)
4          Allison, Master Hudson Trevor
...
1308          Zakarian, Mr Artun
1309          Zakarian, Mr Maprieder
1310          Zenni, Mr Philip
1311          Lievens, Mr Rene
1312          Zimmerman, Leo
Name: Name, Length: 1313, dtype: object
```

Ввод [7]:

```
tit[['Name', 'Sex']].head(7)
```

Out[7]:

	Name	Sex
0	Allen, Miss Elisabeth Walton	female
1	Allison, Miss Helen Loraine	female
2	Allison, Mr Hudson Joshua Creighton	male
3	Allison, Mrs Hudson JC (Bessie Waldo Daniels)	female
4	Allison, Master Hudson Trevor	male
5	Anderson, Mr Harry	male
6	Andrews, Miss Kornelia Theodosia	female

Если же нам нужно вывести не первые n элементов таблицы, а последние, то обратимся к функции tail

Ввод [8]:

```
tit.tail(7)
```

Out[8]:

	PassengerID	Name	PClass	Age	Sex	Survived	SexCode
1306	1307	Zabour, Miss Hileni	3rd	NaN	female	0	1
1307	1308	Zabour, Miss Tamini	3rd	NaN	female	0	1
1308	1309	Zakarian, Mr Artun	3rd	27.0	male	0	0
1309	1310	Zakarian, Mr Maprieder	3rd	26.0	male	0	0
1310	1311	Zenni, Mr Philip	3rd	22.0	male	0	0
1311	1312	Lievens, Mr Rene	3rd	24.0	male	0	0
1312	1313	Zimmerman, Leo	3rd	29.0	male	0	0

Для того чтобы посмотреть каких типов элементы в таблице, воспользуемся функцией dtypes

Ввод [9]:

```
tit.dtypes
```

Out[9]:

```
PassengerID    int64
Name            object
PClass         object
Age            float64
Sex            object
Survived        int64
SexCode         int64
dtype: object
```

Для того чтобы вывести конкретные столбцы с конкретными строками, будем использовать функции `loc` и `iloc`

Ввод [10]:

```
tit.loc[[5,11,20],['Name','Sex']]
```

Out[10]:

	Name	Sex
5	Anderson, Mr Harry	male
11	Astor, Mrs John Jacob (Madeleine Talmadge Force)	female
20	Behr, Mr Karl Howell	male

Ввод [11]:

```
tit.iloc[[5,11,20],[1,4]]
```

Out[11]:

	Name	Sex
5	Anderson, Mr Harry	male
11	Astor, Mrs John Jacob (Madeleine Talmadge Force)	female
20	Behr, Mr Karl Howell	male

Ввод [12]:

```
tit.iloc[5:21,:3]
```

Out[12]:

	PassengerID	Name	PClass
5	6	Anderson, Mr Harry	1st
6	7	Andrews, Miss Kornelia Theodosia	1st
7	8	Andrews, Mr Thomas, jr	1st
8	9	Appleton, Mrs Edward Dale (Charlotte Lamson)	1st
9	10	Artagaveytia, Mr Ramon	1st
10	11	Astor, Colonel John Jacob	1st
11	12	Astor, Mrs John Jacob (Madeleine Talmadge Force)	1st
12	13	Aubert, Mrs Leontine Pauline	1st
13	14	Barkworth, Mr Algernon H	1st
14	15	Baumann, Mr John D	1st
15	16	Baxter, Mrs James (Helene DeLaudeniére Chaput)	1st
16	17	Baxter, Mr Quigg Edmond	1st
17	18	Beattie, Mr Thomson	1st
18	19	Beckwith, Mr Richard Leonard	1st
19	20	Beckwith, Mrs Richard Leonard (Sallie Monypeny)	1st
20	21	Behr, Mr Karl Howell	1st

Что бы вывести элементы таблицы с каким-нибудь условием, то передадим нашему dataset в качестве аргумента булеву маску

Ввод [13]:

```
tit[tit.Age>18]
```

Out[13]:

PassengerID		Name	PClass	Age	Sex	Survived	SexCode
0	1	Allen, Miss Elisabeth Walton	1st	29.0	female	1	1
2	3	Allison, Mr Hudson Joshua Creighton	1st	30.0	male	0	0
3	4	Allison, Mrs Hudson JC (Bessie Waldo Daniels)	1st	25.0	female	0	1
5	6	Anderson, Mr Harry	1st	47.0	male	1	0
6	7	Andrews, Miss Kornelia Theodosia	1st	63.0	female	1	1
...
1308	1309	Zakarian, Mr Artun	3rd	27.0	male	0	0
1309	1310	Zakarian, Mr Maprieder	3rd	26.0	male	0	0
1310	1311	Zenni, Mr Philip	3rd	22.0	male	0	0
1311	1312	Lievens, Mr Rene	3rd	24.0	male	0	0
1312	1313	Zimmerman, Leo	3rd	29.0	male	0	0

630 rows × 7 columns

Что бы вывести элементы, которые подходят под несколько условий мы можем воспользоваться функцией `isin` или использовать несколько булевых масок

Ввод [14]:

```
tit[tit.Age.isin([5,10,18])]
```

Out[14]:

PassengerID		Name	PClass	Age	Sex	Survived	SexCode
201	202	Penasco, Mr Victor de Satode	1st	18.0	male	0	0
220	221	Ryerson, Miss Emily Borie	1st	18.0	female	1	1
236	237	Smith, Mrs Lucien Philip (Mary Eloise Hughes)	1st	18.0	female	1	1
258	259	Taussig, Miss Ruth	1st	18.0	female	1	1
324	325	Andrew, Mr Edgar Samuel	2nd	18.0	male	0	0
329	330	Bailey, Mr Percy Andrew	2nd	18.0	male	0	0
386	387	Deacon, Mr Percy	2nd	18.0	male	0	0
391	392	Dibden, Mr William	2nd	18.0	male	0	0
393	394	Doling, Miss Elsie	2nd	18.0	female	1	1
442	443	Hiltunen, Miss Marta	2nd	18.0	female	0	1
511	512	Nasser (Nasrallah), Mrs Nicholas	2nd	18.0	female	1	1
553	554	Silven, Miss Lyyli	2nd	18.0	female	1	1
601	602	Swane, Mr George	2nd	18.0	male	0	0
608	609	Abraham, Mrs Joseph (Sophie Easu)	3rd	18.0	female	1	1
615	616	Aks, Mrs Sam (Leah Rosen)	3rd	18.0	female	1	1
637	638	Arnold, Mrs Josef (Josephine Frank)	3rd	18.0	female	0	1
642	643	Asplund, Master Carl Edgar	3rd	5.0	male	0	0
647	648	Asplund, Miss Lillian Gertrud	3rd	5.0	female	1	1
661	662	Badman, Miss Emily Louisa	3rd	18.0	female	1	1
666	667	Barbara, Miss Saude	3rd	18.0	female	0	1
675	676	Bjorklund, Ernst Herbert	3rd	18.0	male	0	0
685	686	Bradley, Miss Bridget Delia	3rd	18.0	female	1	1
695	696	Burns, Miss Mary Delia	3rd	18.0	female	0	1
696	697	Cacic, Mr Grego	3rd	18.0	male	0	0
720	721	Chronopoulos, Mr Demetrios	3rd	18.0	male	0	0
787	788	Edvardsson, Mr Gustaf Hjalmar	3rd	18.0	male	0	0
794	795	Emanuel, Miss Virginia Ethel	3rd	5.0	female	1	1
807	808	Ford, Mr Edward Watson	3rd	18.0	male	0	0
829	830	Goodwin, Miss Jessie A	3rd	10.0	female	0	1
857	858	Hegarty, Miss Nora	3rd	18.0	female	0	1
938	939	Klasen, Mr Klas Albin	3rd	18.0	male	0	0
1263	1264	Turja, Miss Anna Sofia	3rd	18.0	female	1	1
1269	1270	Van der Planke, Miss Augusta	3rd	18.0	female	0	1

	PassengerID	Name	PClass	Age	Sex	Survived	SexCode
1276	1277	Van Impe, Miss Catharine	3rd	10.0	female	0	1
1292	1293	Wiklund, Mr Jacob Alfred	3rd	18.0	male	0	0

Ввод [15]:

```
tit[(tit.Age == 5)|(tit.Age == 10)|(tit.Age == 18)]
```

Out[15]:

PassengerID		Name	PClass	Age	Sex	Survived	SexCode
201	202	Penasco, Mr Victor de Satode	1st	18.0	male	0	0
220	221	Ryerson, Miss Emily Borie	1st	18.0	female	1	1
236	237	Smith, Mrs Lucien Philip (Mary Eloise Hughes	1st	18.0	female	1	1
258	259	Taussig, Miss Ruth	1st	18.0	female	1	1
324	325	Andrew, Mr Edgar Samuel	2nd	18.0	male	0	0
329	330	Bailey, Mr Percy Andrew	2nd	18.0	male	0	0
386	387	Deacon, Mr Percy	2nd	18.0	male	0	0
391	392	Dibden, Mr William	2nd	18.0	male	0	0
393	394	Doling, Miss Elsie	2nd	18.0	female	1	1
442	443	Hiltunen, Miss Marta	2nd	18.0	female	0	1
511	512	Nasser (Nasrallah), Mrs Nicholas	2nd	18.0	female	1	1
553	554	Silven, Miss Lyyli	2nd	18.0	female	1	1
601	602	Swane, Mr George	2nd	18.0	male	0	0
608	609	Abraham, Mrs Joseph (Sophie Easu)	3rd	18.0	female	1	1
615	616	Aks, Mrs Sam (Leah Rosen)	3rd	18.0	female	1	1
637	638	Arnold, Mrs Josef (Josephine Frank)	3rd	18.0	female	0	1
642	643	Asplund, Master Carl Edgar	3rd	5.0	male	0	0
647	648	Asplund, Miss Lillian Gertrud	3rd	5.0	female	1	1
661	662	Badman, Miss Emily Louisa	3rd	18.0	female	1	1
666	667	Barbara, Miss Saude	3rd	18.0	female	0	1
675	676	Bjorklund, Ernst Herbert	3rd	18.0	male	0	0
685	686	Bradley, Miss Bridget Delia	3rd	18.0	female	1	1
695	696	Burns, Miss Mary Delia	3rd	18.0	female	0	1
696	697	Cacic, Mr Grego	3rd	18.0	male	0	0
720	721	Chronopoulos, Mr Demetrios	3rd	18.0	male	0	0
787	788	Edvardsson, Mr Gustaf Hjalmar	3rd	18.0	male	0	0
794	795	Emanuel, Miss Virginia Ethel	3rd	5.0	female	1	1
807	808	Ford, Mr Edward Watson	3rd	18.0	male	0	0
829	830	Goodwin, Miss Jessie A	3rd	10.0	female	0	1
857	858	Hegarty, Miss Nora	3rd	18.0	female	0	1
938	939	Klasen, Mr Klas Albin	3rd	18.0	male	0	0
1263	1264	Turja, Miss Anna Sofia	3rd	18.0	female	1	1

PassengerID		Name	PClass	Age	Sex	Survived	SexCode
1269	1270	Van der Planke, Miss Augusta	3rd	18.0	female	0	1
1276	1277	Van Impe, Miss Catharine	3rd	10.0	female	0	1
1292	1293	Wiklund, Mr Jacob Alfred	3rd	18.0	male	0	0

Ввод [16]:

```
tit[(tit.Age == 5)|(tit.Age == 10)|(tit.Age == 18)] == tit[tit.Age.isin([5,10,18])]
```

Out[16]:

	PassengerID	Name	PClass	Age	Sex	Survived	SexCode
201	True	True	True	True	True	True	True
220	True	True	True	True	True	True	True
236	True	True	True	True	True	True	True
258	True	True	True	True	True	True	True
324	True	True	True	True	True	True	True
329	True	True	True	True	True	True	True
386	True	True	True	True	True	True	True
391	True	True	True	True	True	True	True
393	True	True	True	True	True	True	True
442	True	True	True	True	True	True	True
511	True	True	True	True	True	True	True
553	True	True	True	True	True	True	True
601	True	True	True	True	True	True	True
608	True	True	True	True	True	True	True
615	True	True	True	True	True	True	True
637	True	True	True	True	True	True	True
642	True	True	True	True	True	True	True
647	True	True	True	True	True	True	True
661	True	True	True	True	True	True	True
666	True	True	True	True	True	True	True
675	True	True	True	True	True	True	True
685	True	True	True	True	True	True	True
695	True	True	True	True	True	True	True
696	True	True	True	True	True	True	True
720	True	True	True	True	True	True	True
787	True	True	True	True	True	True	True
794	True	True	True	True	True	True	True
807	True	True	True	True	True	True	True
829	True	True	True	True	True	True	True
857	True	True	True	True	True	True	True
938	True	True	True	True	True	True	True
1263	True	True	True	True	True	True	True
1269	True	True	True	True	True	True	True
1276	True	True	True	True	True	True	True

PassengerID	Name	PClass	Age	Sex	Survived	SexCode
1292	True	True	True	True	True	True

Что бы посчитать количество непустых элементов используем функцию `notna` , а для количества пустых `isna`

Ввод [17]:

```
tit.Age.notna()
```

Out[17]:

```
0      True
1      True
2      True
3      True
4      True
...
1308   True
1309   True
1310   True
1311   True
1312   True
Name: Age, Length: 1313, dtype: bool
```

Ввод [18]:

```
tit[tit.Age.notna()]
```

Out[18]:

PassengerID		Name	PClass	Age	Sex	Survived	SexCode
0	1	Allen, Miss Elisabeth Walton	1st	29.00	female	1	1
1	2	Allison, Miss Helen Loraine	1st	2.00	female	0	1
2	3	Allison, Mr Hudson Joshua Creighton	1st	30.00	male	0	0
3	4	Allison, Mrs Hudson JC (Bessie Waldo Daniels)	1st	25.00	female	0	1
4	5	Allison, Master Hudson Trevor	1st	0.92	male	1	0
...
1308	1309	Zakarian, Mr Artun	3rd	27.00	male	0	0
1309	1310	Zakarian, Mr Maprieder	3rd	26.00	male	0	0
1310	1311	Zenni, Mr Philip	3rd	22.00	male	0	0
1311	1312	Lievens, Mr Rene	3rd	24.00	male	0	0
1312	1313	Zimmerman, Leo	3rd	29.00	male	0	0

756 rows × 7 columns

Ввод [19]:

```
tit.Age.isna().sum()
```

Out[19]:

557

Ввод [20]:

```
tit.loc[tit.Age.notna(), 'Name']
```

Out[20]:

```
0          Allen, Miss Elisabeth Walton
1          Allison, Miss Helen Loraine
2          Allison, Mr Hudson Joshua Creighton
3  Allison, Mrs Hudson JC (Bessie Waldo Daniels)
4          Allison, Master Hudson Trevor
...
1308          Zakarian, Mr Artun
1309          Zakarian, Mr Maprieder
1310          Zenni, Mr Philip
1311          Lievens, Mr Rene
1312          Zimmerman, Leo
Name: Name, Length: 756, dtype: object
```

Для сортировки таблицы по какому-нибудь атрибуту или нескольким атрибутам используем функцию `sort_values`. Если же хотим сортировку в обратном порядке, то добавим ключ `ascending` с параметром `False`, для прямой `True`

Ввод [21]:

```
tit.sort_values('Age')
```

Out[21]:

	PassengerID	Name	PClass	Age	Sex	Survived	SexCode
763	764	Dean, Miss Elizabeth Gladys (Millvena)	3rd	0.17	female	1	1
751	752	Danbom, Master Gilbert Sigvard Emanuel	3rd	0.33	male	0	0
544	545	Richards, Master George Sidney	2nd	0.80	male	1	0
616	617	Aks, Master Philip	3rd	0.83	male	1	0
358	359	Caldwell, Master Alden Gates	2nd	0.83	male	1	0
...
1300	1301	Wiseman, Mr Phillippe	3rd	NaN	male	0	0
1302	1303	Yalsevac, Mr Ivan	3rd	NaN	male	1	0
1305	1306	Youssef, Mr Gerios	3rd	NaN	male	0	0
1306	1307	Zabour, Miss Hileni	3rd	NaN	female	0	1
1307	1308	Zabour, Miss Tamini	3rd	NaN	female	0	1

1313 rows × 7 columns

Ввод [22]:

```
tit.sort_values('Age').head(22)
```

Out[22]:

PassengerID		Name	PClass	Age	Sex	Survived	SexCode
763	764	Dean, Miss Elizabeth Gladys (Millvena)	3rd	0.17	female	1	1
751	752	Danbom, Master Gilbert Sigvard Emanuel	3rd	0.33	male	0	0
544	545	Richards, Master George Sidney	2nd	0.80	male	1	0
616	617	Aks, Master Philip	3rd	0.83	male	1	0
358	359	Caldwell, Master Alden Gates	2nd	0.83	male	1	0
4	5	Allison, Master Hudson Trevor	1st	0.92	male	1	0
339	340	Becker, Master Richard F	2nd	1.00	male	1	0
478	479	LaRoche, Miss Louise	2nd	1.00	female	1	1
893	894	Johnson, Miss Eleanor Ileen	3rd	1.00	female	1	1
425	426	Hamalainen, Master Viljo	2nd	1.00	male	1	0
762	763	Dean, Master Bertram Vere	3rd	1.00	male	1	0
936	937	Klasen, Miss Gertrud Emilia	3rd	1.50	female	0	1
1188	1189	Sandstrom, Miss Beatrice Irene	3rd	1.50	female	0	1
627	628	Andersson, Miss Ellis Anna Maria	3rd	2.00	female	0	1
866	867	Hirvonen, Miss Hildur E	3rd	2.00	female	0	1
512	513	Navratil, Master Edmond Roger	2nd	2.00	male	1	0
1	2	Allison, Miss Helen Loraine	1st	2.00	female	0	1
537	538	Quick, Miss Phyllis May	2nd	2.00	female	1	1
491	492	Mallet, Master Andre	2nd	2.00	male	1	0
585	586	Wells, Master Ralph Lester	2nd	2.00	male	1	0
546	547	Richards, Master William Rowe	2nd	3.00	male	1	0
514	515	Navratil, Master Michel M	2nd	3.00	male	1	0

Ввод [23]:

```
tit.sort_values(['Age','Name'], ascending = [False,True])
```

Out[23]:

	PassengerID	Name	PClass	Age	Sex	Survived	SexCode
9	10	Artagaveytia, Mr Ramon	1st	71.0	male	0	0
119	120	Goldschmidt, Mr George B	1st	71.0	male	0	0
505	506	Mitchell, Mr Henry Michael	2nd	71.0	male	0	0
72	73	Crosby, Captain Edward Gifford	1st	70.0	male	0	0
73	74	Crosby, Mrs Edward Gifford (Catherine Elizabet...	1st	69.0	female	1	1
...
597	598	Yrois, Miss Henriette	2nd	NaN	female	0	1
1306	1307	Zabour, Miss Hileni	3rd	NaN	female	0	1
1307	1308	Zabour, Miss Tamini	3rd	NaN	female	0	1
387	388	de Brito, Mr Jose Joaquim	2nd	NaN	male	0	0
81	82	de Villiers, Madame Berthe	1st	NaN	female	1	1

1313 rows × 7 columns

Ввод [24]:

```
tit2 = tit.copy(deep=True)#конуруем наш dataset
```

Ввод [25]:

```
tit.count()# выводим количество непустых элементов
```

Out[25]:

```
PassengerID    1313
Name            1313
PClass         1313
Age             756
Sex            1313
Survived       1313
SexCode        1313
dtype: int64
```

Ввод [26]:

```
tit.Age.mean(),tit.Age.median()#выводим среднее и медиану
```

Out[26]:

```
(30.397989417989415, 28.0)
```

Ввод [27]:

```
tit.Age.describe()
```

Out[27]:

```
count      756.000000
mean       30.397989
std        14.259049
min         0.170000
25%        21.000000
50%        28.000000
75%        39.000000
max        71.000000
Name: Age, dtype: float64
```

Ввод [28]:

```
tit.describe()
```

Out[28]:

	PassengerID	Age	Survived	SexCode
count	1313.000000	756.000000	1313.000000	1313.000000
mean	657.000000	30.397989	0.342727	0.351866
std	379.174762	14.259049	0.474802	0.477734
min	1.000000	0.170000	0.000000	0.000000
25%	329.000000	21.000000	0.000000	0.000000
50%	657.000000	28.000000	0.000000	0.000000
75%	985.000000	39.000000	1.000000	1.000000
max	1313.000000	71.000000	1.000000	1.000000

Ввод [29]:

```
tit.groupby('Sex')['Age'].mean()
```

Out[29]:

```
Sex
female    29.396424
male      31.014338
Name: Age, dtype: float64
```


Ввод [30]:

```
tit.groupby(['Sex', 'Survived'])['Age'].agg(['mean', 'median'])
```

Out[30]:

		mean	median
Sex	Survived		
female	0	24.901408	24.0
	1	30.867143	29.0
male	0	32.320780	29.0
	1	25.951875	26.0

Ввод [31]:

```
tit.Sex.value_counts()
```

Out[31]:

```
male      851
female    462
Name: Sex, dtype: int64
```

Ввод [32]:

```
tit.corr()
```

Out[32]:

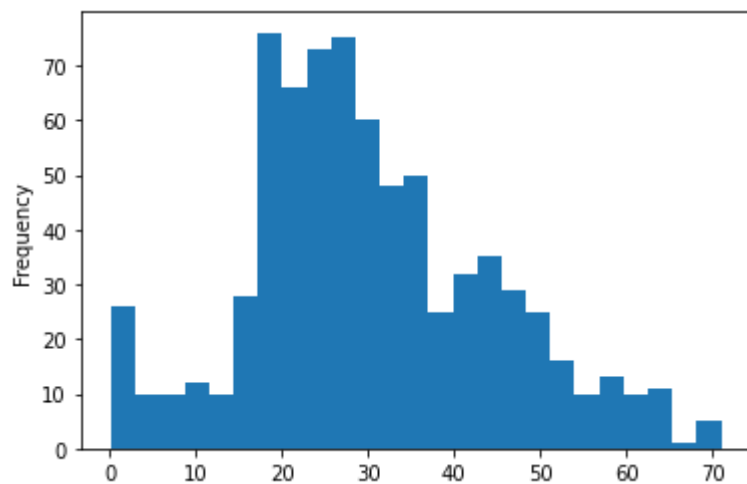
	PassengerID	Age	Survived	SexCode
PassengerID	1.000000	-0.340922	-0.352925	-0.125035
Age	-0.340922	1.000000	-0.061254	-0.055138
Survived	-0.352925	-0.061254	1.000000	0.502891
SexCode	-0.125035	-0.055138	0.502891	1.000000

Ввод [33]:

```
tit.Age.plot(kind='hist',bins=25)#строим гистограмму по возрасту
```

Out[33]:

<AxesSubplot:ylabel='Frequency'>

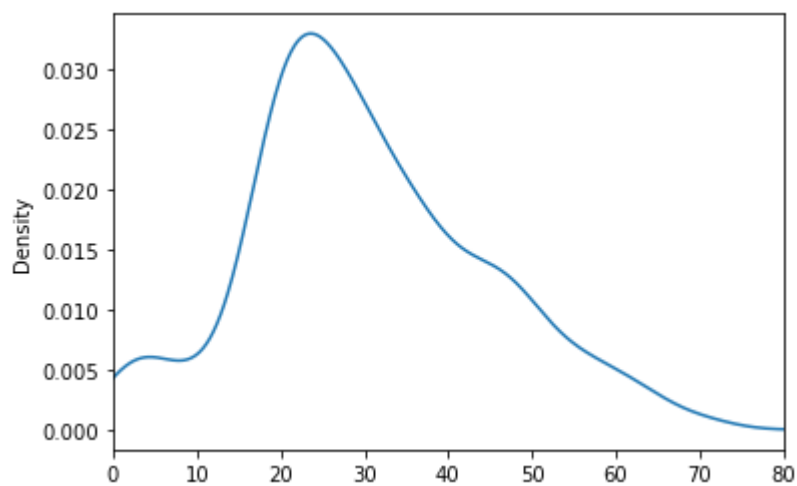


Ввод [34]:

```
tit.Age.plot(kind='kde',xlim=[0,80])#аппроксимация возраста
```

Out[34]:

<AxesSubplot:ylabel='Density'>



Ввод [35]:

```
tit.groupby('Sex')['Age'].plot(kind='kde',xlim=[0,80],legend=True)#аппроксимация возраста n
```

Out[35]:

Sex
female AxesSubplot(0.125,0.125;0.775x0.755)
male AxesSubplot(0.125,0.125;0.775x0.755)
Name: Age, dtype: object

