assi1-a-pandas-1

October 21, 2024

[ ]:

```
[ ]: import pandas as pd
     dict1 = {
          "Names" : ['Adhya','sakshi','ishwar', 'tejas','siddhesh'],
     "city" : ['akot','dhule','sambhajinagar','ratnagiri','pune'],
     "marks" : [90,87,94,98,99]
     }
     data = pd.DataFrame(dict1)
     data
```

[ ]:

	Names	city	marks
0	Adhya	akot	90
1	sakshi	dhule	87
2	ishwar	sambhajinagar	94
3	tejas	ratnagiri	98
4	siddhesh	pune	99

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

```
[ ]: data.to_csv('friends.csv')
```

```
[ ]: data.to_csv('friends_index_false.csv',index=False)
```

```
[ ]: data.head(2)
```

[ ]:

	Names	city	marks
0	Adhya	akot	90
1	sakshi	dhule	87

```
[ ]: data.tail(3)
```

[ ]:

	Names	city	marks
2	ishwar	sambhajinagar	94
3	tejas	ratnagiri	98
4	siddhesh	pune	99

```
[ ]: print(pd.options.display.max_rows)
```

```
60
```

```
[ ]: print(data.to_string())
```

```
   Names          city  marks
0  Adhya          akot     90
1  sakshi         dhule    87
2  ishwar  sambhajinagar    94
3   tejas       ratnagiri   98
4  siddhesh         pune    99
```

```
[ ]: data.loc[3]
```

```
[ ]: Names      tejas
     city     ratnagiri
     marks           98
     Name: 3, dtype: object
```

```
[ ]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5 entries, 0 to 4
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Names   5 non-null      object
 1   city    5 non-null      object
 2   marks   5 non-null      int64
dtypes: int64(1), object(2)
memory usage: 248.0+ bytes
```

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

```
[ ]:        marks
count    5.000000
mean    93.600000
std      5.128353
min     87.000000
25%     90.000000
50%     94.000000
75%     98.000000
max     99.000000
```

```
[ ]: data.size
```

```
[ ]: 15
```

```
[ ]: data['Names'].dtype
```

[ ]: dtype('O')

[ ]: data.index = ['first','second','third','fourth','fifth']

[ ]: data

[ ]:

	Names	city	marks
first	Adhya	akot	90
second	sakshi	dhule	87
third	ishwar	sambhajinagar	94
fourth	tejas	ratnagiri	98
fifth	siddhesh	pune	99

[ ]:
```
dic = {"name" :['python','C','java','C++'],
       "por":[10,223,43,49],
       "rank":[1,2,3,4]
       }
var = pd.Series(dic)
print(var)
print(type(var))
print(var[2])
```

```
name      [python, C, java, C++]
por              [10, 223, 43, 49]
rank                   [1, 2, 3, 4]
dtype: object
<class 'pandas.core.series.Series'>
[1, 2, 3, 4]
```

```
<ipython-input-22-0e59dc8d5c77>:8: FutureWarning: Series.__getitem__ treating
keys as positions is deprecated. In a future version, integer keys will always be
treated as labels (consistent with DataFrame behavior). To access a value by
position, use `ser.iloc[pos]`
  print(var[2])
```

[ ]: newdf = pd.DataFrame(np.random.rand(334))

[ ]: newdf

[ ]:

	0
0	0.911242
1	0.546549
2	0.980070
3	0.839567
4	0.228103
..	...
329	0.552858
330	0.660450

```
331 332 333                                  0
                                             .
                                             9
```

[334 rows x 1 columns]

[ ]: type(newdf)

[ ]:

pandas.core.frame.DataFrame

[ ]: df df sv('friend s.csv')

[ ]:

	Unnamed: 0	Names	city	marks
0	0	Adhya	akot	90
1	1	sakshi	dhule	87
2	2	ishwar	sambhajinagar	94
3	3	tejas	ratnagiri	98
4	4	siddhesh	pune	99

[ ]: data.size

[ ]: 15

[ ]: data['Names'].isnull().sum()

[ ]: 0

[ ]:
```
df1 = pd.read_csv('friends.csv')

new_df1 = df1.dropna()
print(new_df1.to_string())
```

```
  Unnamed: 0     Names           city  m
0          0     Adhya           akot
1          1    sakshi          dhule
2          2    ishwar  sambhajinagar
3          3     tejas      ratnagiri
4          4  siddhesh           pune
```

```
df1.dropna(inplace = Tr ue)

print(df1.to_string())
```

```
  Unnamed: 0     Names           city  marks
0          0     Adhya           akot     90
1          1    sakshi          dhule     87
2          2    ishwar  sambhajinagar     94
3          3     tejas      ratnagiri     98
4          4  siddhesh           pune     99
```

[ ]:

```
[ ]: import pandas as pd
     mydata = {
           'Language' : ['C++','C','Java','Python','Kotlin'],
     'subject' : ['oops','cpps','java','cpps2','anroid'],
     'credit': [4,6,12,7,2]
     }
     set = pd.DataFrame(mydata)
     set
```

[ ]:

	Language	subject	credit
0	C++	oops	4
1	C	cpps	6
2	Java	java	12
3	Python	cpps2	7
4	Kotlin	anroid	2

```
[ ]: print(set[['Language','subject']])
```

	Language	subject
0	C++	oops
1	C	cpps
2	Java	java
3	Python	cpps2
4	Kotlin	anroid

```
[ ]: variable = set

print(variable)
set['subject',1]= 'DSA'
print(set)
```

	Language	subject	credit	(1, subject)	(subject, 1)
0	C++	oops	4	DSA	DSA
1	C	cpps	6	DSA	DSA
2	Java	java	12	DSA	DSA
3	Python	cpps2	7	DSA	DSA
4	Kotlin	anroid	2	DSA	DSA

	Language	subject	credit	(1, subject)	(subject, 1)
0	C++	oops	4	DSA	DSA
1	C	cpps	6	DSA	DSA
2	Java	java	12	DSA	DSA
3	Python	cpps2	7	DSA	DSA
4	Kotlin	anroid	2	DSA	DSA

```
[ ]: row1 = set.loc[3]

     row2 =
```

[ ]: Language

```
                    True
subject             True
credit              True
(1, subject)        True
(subject, 1)        True
Name: 3, dtype: bool
```

[ ]: