10. Pandas Search, Join, Split, Rearrange
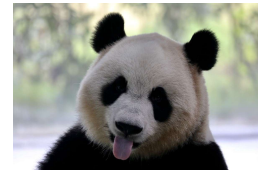
PYTHON COURSE

SIN YONG TENG

1

Pandas: Functions for search

Function	Purpose
df.lookup(x)	Use a column to look up values
df.query(x)	Search variables based on simple statements
df.where(x)	Returns DataFrame with values that satisfy condition.
df.isin(x)	Check whether a value is in the DataFrame

2

Pandas "Is In?" Function

```python
import pandas as pd

data = {'A': [1,2,3],
        'B': [4,5,6],
        'C': [7,8,9]}


df = pd.DataFrame(data, columns = ['A','B', 'C'], index=['First', 'Second', 'Third'])

print(df.isin([1,5,9]))

print(df.isin({'A':[2,3]}))

data2 = {'A': [1,2],
         'B': [4,5]}

df2 = pd.DataFrame(data2, columns = ['A','B'], index=['First', 'Second'])

print(df.isin(df2))
```

3

Pandas Where

```python
import pandas as pd

data = {'A': [1,2,3],
        'B': [4,5,6],
        'C': [7,8,9]}


df = pd.DataFrame(data, columns = ['A','B', 'C'], index=['First', 'Second', 'Third'])

print(df.where(df<5,-df))

print(df.where(df.isin([1,3,5]),-df))
```

4

Pandas Query

```python
import pandas as pd

data = {'A': [1,2,3],
        'B': [4,5,6],
        'C': [7,8,9]}

df = pd.DataFrame(data, columns = ['A','B', 'C'], index=['First', 'Second', 'Third'])

print(df.query("(A+B)%3==0"))
print(df.query("(A+B)%3!=0"))
print(df.query("A*B*C<=80"))
```

5

Pandas Lookup

```python
import pandas as pd

data = {'A': [1,2,3],
        'B': [4,5,6],
        'C': [7,8,9],
        'Best':['A','C','A']}

df = pd.DataFrame(data, columns = ['Best','A','B', 'C'], index=['First', 'Second', 'Third'])

df['value'] = df.lookup(df.index, df['Best'])

print(df)
```

6

Challenge 1: Search Functions

You are given a DataFrame:

	A	B	C
First	1	4	7
Second	2	5	8
Third	3	6	9

1. Change all odd values to zero.
2. Remove rows (first, second, third) where the sum of the value in row is divisible by 4.
3. Add a "Best" column with ["A","A", "C"] with index=['Second', 'Third'], then look up the best values.

What did you get?

7

Pandas Drop, Concatenate, Reindex

Function	Purpose
df.drop(label, axis)	Split and remove specific data in DataFrames
df.concat([df1,df2,df3])	Combine DataFrames
df.reindex(label, axis)	Rearrange the DataFrames by index

8

Examples: Drop, Concatenate, Reindex

```python
import pandas as pd
data = {'A': [1,2,3],
        'B': [4,5,6],
        'C': [7,8,9]}

df = pd.DataFrame(data, columns = ['A','B', 'C'], index=['First', 'Second', 'Third'])
print(df)

#Drop
df1=df.drop(['First','Second'],axis=0)
print(df1)

df2=df.drop(['A','B'],axis=1)
print(df2)

#Concatenate
df3=pd.concat([df1,df2],axis=0)
print(df3)

#Reindex
df4=df.reindex(["C","B","A"],axis=1)
print(df4)

df5=df.reindex(["Second","Third","First"],axis=0)
print(df5)
```

9

Challenge 2: DataFrame Manipulation

You are given a DataFrame:

	A	B	C
First	1	4	7
Second	2	5	8
Third	3	6	9
Forth	4	7	10

1. Drop the "second" row.
2. If any values in the DataFrame is Even, divide it by 2.
3. Add another column named "D", with incrementing value, i.e. [1,2,3,…]
4. Reindex the row to be in order, i.e. [First, Second, Thirds, …]. The column should be reversed in order, i.e. [D,C,B,A]
5. Replace all "NaN" values by 0. (Hint: df.fillna(0))

10

HOMEWORK: The Iris Dataset

1. Download the open-sourced Iris Dataset: https://gist.github.com/netj/8836201

2. Separate the variety of flower: Setosa, Versicolor, Virginica. Drop all Virginica data.

3. Calculate the Approximate Area of Sepal: sepal.area= sepal.length x sepal.width

4. Calculated the Approximated Area of petal: petal.area= petal.length x petal.width

5. Find the mean value of sepal area and petal area for Setosa and Versicolor.
   Hint: df.mean()

6. Compared the sum of area for both flower. Which is larger?
(sum of area=average(sepal.area)+average(petal.area))

11

Conclusion

1. Function for search
2. Is in function
3. Where function
4. Query function
5. Lookup function
6. Exercise
7. Drop, Concatenate, Reindex
8. Example
9. Dataframe manipulation exercise
10. Iris Dataset Homework

12