# Explain three dimensional data indexing?

## Three dimensional index clusters entries so as to exploit “nearness” in multidimensional space.

## The basic motivation for multidimensional indexing is that for efficient content based retrieval It is necessary to cluster entitles so as to exploit nearness in multidimensional space

## \*most multidimensional index structures:

-k-d trees

-Point Quadtrees

-R,R\*,R+ Trees

-SS,SR trees

-M trees

# What is the difference between series and data frame?

# Series

## A pandas series is a one dimensional data structure that comprises of a key value pair.

## It is similar to python dictionary. Except it provides more freedom to manipulate and edit the data.

## To initialize a series we use pd. series()

Example:

Import pandas as pd

####INITIALIZATION####

#STRING SERIES

Fruits = pd. Series ([“apples” , ”oranges”, ”bananas”])

Print (“MY FRUIT SERIES”)

Print (Fruits,”\n”)

#FLOAT SERIES

Temperature=pd. Series ([1.2,4.6,12]), index= [“a”, ”b”, ”c”, ”d”])

Print (TEMPERATURE IN CELSIUS”)

Print (temperature, ”\n”)

#INTEGER SERIES

Factors\_ of\_ 12 = pd. Series ([1.2,2.4,2])

Print (“FACTORS OF 12 “)

Print (Factors\_of\_12,”\n”)

####QUERY####

#USING INDEX

Print (“2nd fruit: “ , fruits. iloc [1])

#OR

Print (2nd fruit: “, fruits [1] , ”\n”)

#USING KEY

Print(“temperature at key \ “b\”: ”,temperature. loc[ “b”])

## Data frame:-

## A pandas data frame is a two dimensional data structure that can be thought as a spreadsheet. A data can also be thought of as a combination of two or more series.

## To initialize a data frame use pd. Data frame:

Example:

Import pandas as pd

####INITIALIZATION####

Fruits\_ jack = [“apples”, ”oranges”, ”bananas”]

Fruits\_ john = [“guavas”, “kiwis”, “strawberries”]

Index= [“a”, “b”, “c”]

All \_fruits = {“jack’s: fruits\_ jack. “John’s “: fruits\_ john}

Fruits=pd. Data frame (all\_ fruits, index=index)

Print ( fruits ,”\n”)

new\_ fruits= fruits.reset\_index(drop=true)

Print (new\_ fruits ,”\n”)

####QUERY####

#USING INDEX

Print (“1nd fruit: “)

Print (“new\_fruits. Iloc [0], “\n”)

#USING KEY

Print (“fruits at key \ “c\”:”)

Print (fruits .loc [“c”], “\n”)

#USING COLUMN NAME

Print(“jack’s fruits: “)

Print(fruits[“jack’s”], “\n”)

#CHAINED QUERY

Print(“john’s third fruit: ”)

Print(new\_ fruits[john’s”] [2], “\n”)

# What role does pandas play in data cleaning?

## Pandas is an open source library that is used for data manipulation and analysis

## It provides many functions and methods to speed up the data analysis process.

## Pandas is built on top of the NumPy package, .hence it takes a lot of basic inspiration from it.

# How do you use Pandas to make a data frame out of n dimensional arrays?

## Import Pandas as pd #Create the Data frame df = pd.Data frame (numpy\_ array)

## Df=pd.Dataframe(numpy\_array,index=[‘day1’,’day2’,’day3’,’day4’],coloumns[‘digits’,’words’])

Example:-

Import pandas as pd

Pd. Data Frame (NumPy\_ array, coloumns = [’Digits’,’Words’])

# Explain the notion of pandas plotting?

## import pandas as pd

## import matplotlib.pylot as plt

## df = pd.read\_csv(‘Data.csv’)

## df. plot()

plt.show()