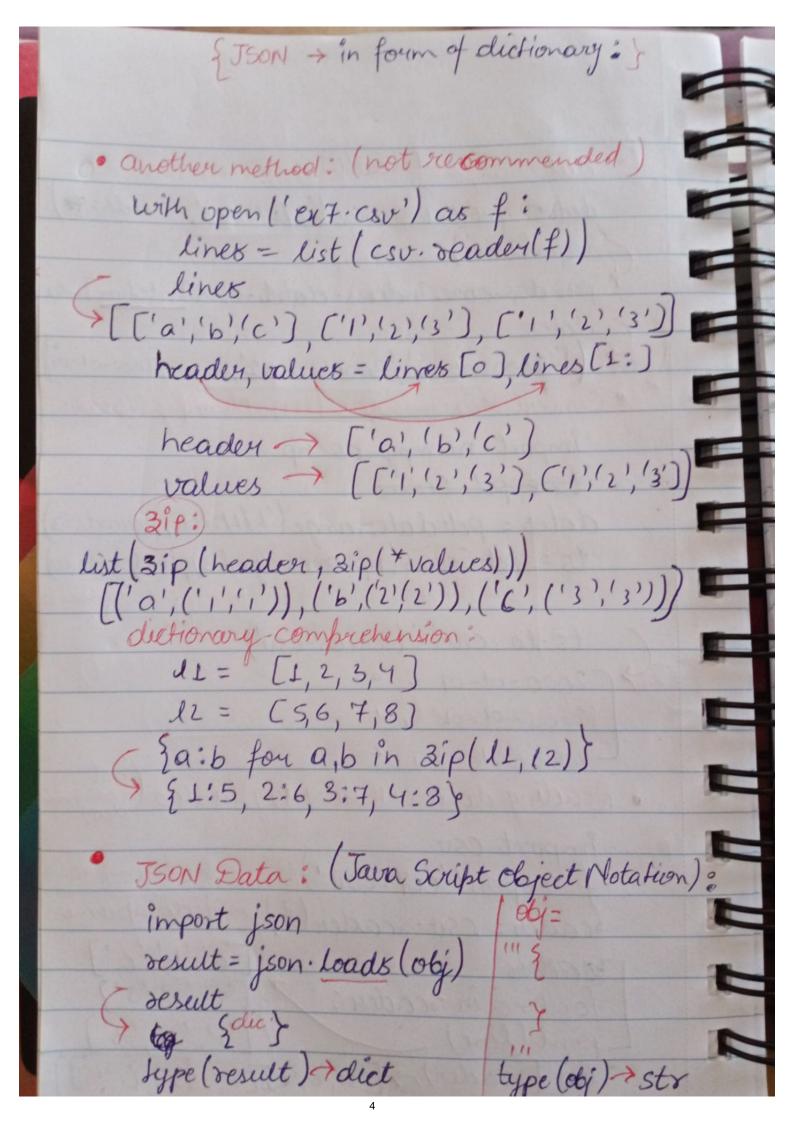


(\*) date-range: (inbuilt) dates = pd. date\_sange ('1/1/2000', Periods=40) type (dates) pandas core indexes datetimes Datetime Index dates [['2000-01-01', '2000-01,02', ---, '2000-01-04] · Convert this in series: ( use Numby lit import numpy as no ts pd Sedes dates = pd. date\_range ('1/1/2000', periods=3 ts = pd. Series (np. arrange (3), index = dates ts.to\_csv('tseries.csv') 2000-01-02 2000-01-03, · Reading data without using Pandas (second import csv f = open ('ex7.csv') reader = csv. reader(f) datatrame contactor for line in reader: - print (line) [ list ( scader) > [['a','b','c'],[''','1','3'],['','1



(str data > json file -> data So, we changed from alleger str -> dict Now, was diet -> 1sts. (s as json = json · dumps (vesult) type (asjson) sto converting dict of join file into dataframe:

- result ['siblings' f (first import & load join) - 12 rame: A, B, C {'A':1, 'B':2, 'C':[3,4,5]}, {'A':6, 'B':7, 'C':[8,9,10]} Siblings = pd. DataFrame (desult ['siblings'], Columns = ['A', 'B', 'C'] - Siblings = 0 1 2 [3,4,5) 1 6 7 [8,9,10] · if you devetly has joon-data, then how to seed? olata = pd. read\_json ('ex7. json') data -> datatrame store data in jon: data to-json()
( Exclosions ( Print ( data to-json())

2 dict y. (with some changes) print (data to - json Occient = 12

