Answers 2) Assumption and Tradeoffs-:

1) I have assumed the shape of each city to be circular for ease of calculation and to get get a thresh hold value. What I thought was to make make clusters of city where urban population lies and see whch all stations are covered in it. Though the calculations are easy but they will not be accurate.

2) I have added a column population_density per sq km. I have searched the population density of cities online and I am assuming they are correct.

I have also noted the assumption and given the logic behind finding the thresholds in the jupyter notebook.

Answer 3) The complexity of mapping function is O(n^2). It takes 10-15 mins to run. If the data is increased it will take more time. It can improved by adding continue statement so that the loop dosen't check for duplicate stations. This was one possible way. Fetching techniques can also be changed in case of bigger dataset. Functions like read_csv takes time. If we are using s3 data lake of AWS, we can use AWSWrangler.

The basic logic (given in the code) of forming clusters (radius threshold) and checking which all Stations come within the clusters should work fine. The complexity can be improved as explained above.

Also we can convert the structured data into unstructured data (json) and use our analytics on it to save time and efficiently use the data.

How to Run the program-:

In order to run the funtion. Please install Jupyter notebook, python 3 and anaconda on your system. I am providing .ipynb file. Please run (Shift + Enter) the cells **sequentially**. It should display the result appropriately. Please make sure you run the cells which fetches the data from the csv files. When you will sequentially come to display_result function and run that cell- it will ask for the date and give the mean and median. Thanks!