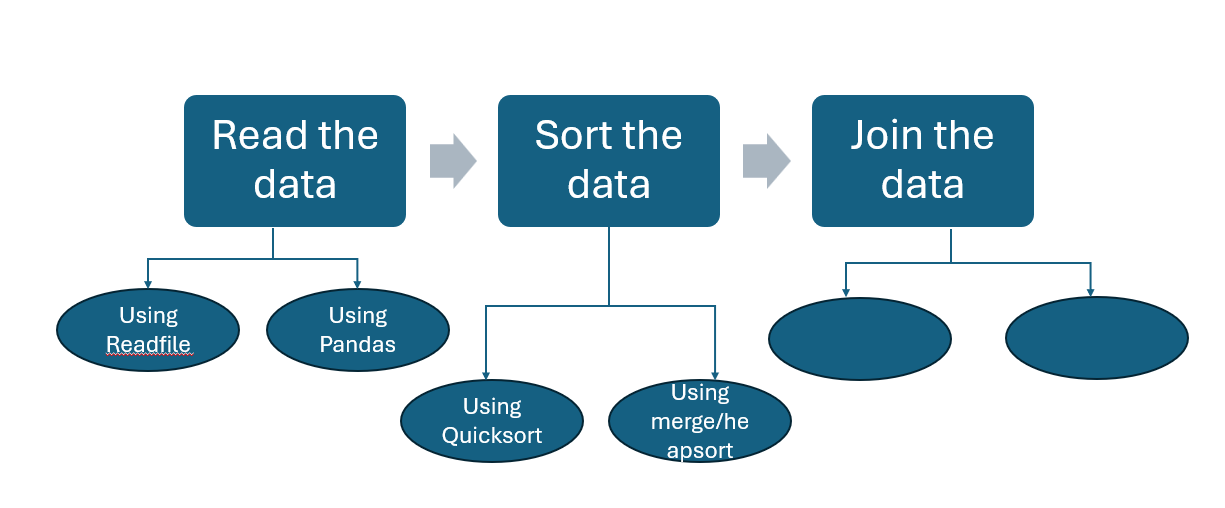
1. Figuring out the required columns
   1. Region – Area.txt + Electricity.txt (on area\_id)
   2. Area – Area.txt + Electricity.txt (on area\_id)
   3. Dwelling – Dwelling.txt + Electricity.txt (on Dwelling\_type\_id)
   4. Year – DateDim.txt + Electricity.txt (on Date\_Id)
   5. Month - DateDim.txt + Electricity.txt (on Date\_Id)
   6. Avg electricity usage – Electricity.txt
2. Two ways to approach:
   1. Pandas Df
   2. Nested dictionary
3. For joining:
   1. Used left join as the electricity data just contains key values.
4. For sorting:
   1. Tried sorting the dfs on the basis of matching keys before merging them.

Interesting points to note:

1. The pandas df uses quick sort, mergesort & heapsort in the background which can be changed (default is quicksort)

Thinkin g process:



1. 2 ways to read the files: one dict inside the list (found better adter the research)
2. Pandas df
3. Started with 2 functiions to compare the reading speed:
4. Join the dictionary, start with joining the smallest lists such as dwelling, area\_id first.

TODO:

* Compare the pandas with/without sort
* Beautify the timeits
* Compare the sort thing in without pandas
* Compare both with and without pandas merge

References:

1. <https://realpython.com/pandas-sort-python/>