NOTE:- Before moving ahead compile all the files.

To compile files do

```
java *.java
```

Inside src folder

```
cd com/company
```

```
java *.java
```

There are two stages to building a tree file:
1. Building a heap file from the provided .csv file
2. Building the tree from the heap file created in 1st stage.

HEAP FILE:

There are two types of B+tree Indexes that can be made using the provided code:

1. Clustered Index
2. Unclustered Index

Experiments done using two different heaps highlighted advantages and disadvantages of both.

Note: The data file needs to be renamed data.csv. (File provided in this folder)

1. UNCLUSTERED INDEX PREPROCESSING

```
java dbload -p 4096 data.csv
```

After this code finishes processing we will have a “heap.pagesize” file.

The code above will make heap.4096 file

Note:- the heap.4096 file will be created in the same folder (src)

BUILDING UNCLUSTERED B+TREE INDEX

Use the following command to run the tree:

```
java com.company.treeQuery 4096 heap.4096
```

Once the tree is constructed the prompt will ask about which index structure we are making? (CLUSTERED/UNCLUSTERED).

Since this part is assuming unclustered you will select “U”

Format of a query:
1- An equality query on SID_Name should be of form

18-11/01/2019 06:00:00 PM

2- A range query on SID_Name should be of form

18-11/01/2019 06:00:00 PM,18-11/01/2019 10:00:00 PM

NOTE : Before going ahead delete heap.4096 from the previous stage if exist.

2. CLUSTERED INDEX PREPROCESSING

To make a clustered index using the provided code we will have to sort the record according to desired order of the key and write that file in binary format using bulk loading approach.

There is a python script provided (sortData.py) which can be used to sort the data with accordance of SID_NAME key.

To make a sorted heap file follow the following steps: 1. Install pandas

```
sudo pip install pandas
```

2. Run sortData.py

```
python sortData.py
```

After the script has stopped processing we will receive a “clustered.csv” file.

Note: make sure that the input file and sortData.py are in the same directory. Note: make sure the data file that is input to sortData.py is named data.csv

Once we have the “clustered.csv” file we can run our dbload to write the binary file.

```
java dbload -p 4096 clustered.csv
```

After this code finishes processing we will have a “heap.pagesize” file. Which will contain all the records in the binary format and can be used to create clustered indexes.

The code above will make heap.4096 file which is made using sorted file.

Note:- the heap.4096 file will be created in the same folder (src)

BUILDING UNCLUSTERED B+TREE INDEX

Use the following command to run the tree:

```
java com.company.treeQuery 4096 heap.4096
```

Once the tree is constructed the prompt will ask about which index structure we are making? (CLUSTERED/UNCLUSTERED).

Since this part is assuming clustered you will select “C”

Format of a query:
1- An equality query on SID_Name should be of form

18-11/01/2019 06:00:00 PM

2- A range query on SID_Name should be of form

18-11/01/2019 06:00:00 PM,18-11/01/2019 10:00:00 PM

NOTE:

FORMATTED RANGE QUERIES THAT YOU CAN COPY PASTE

1-05/01/2009 12:00:00 AM,1-05/01/2009 06:00:00 AM

1-05/01/2009 12:00:00 AM,1-05/05/2009 03:00:00 AM

1-05/01/2009 12:00:00 AM ,1-06/11/2009 04:00:00 PM