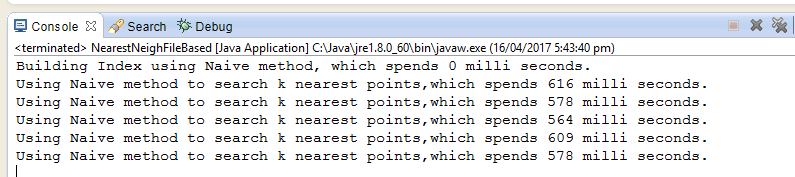
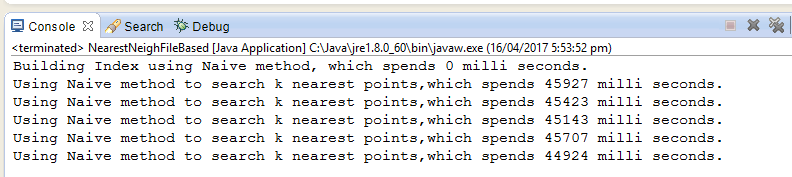
**Scenario 1**

**Using naive method to search 1,10,100,300,500 nearest points respectively.**

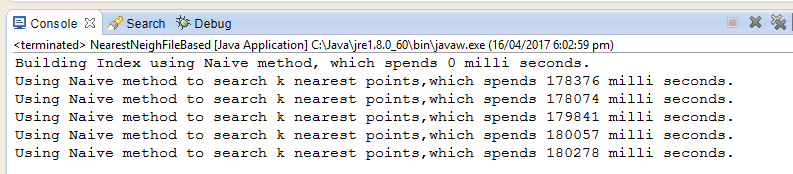
**Under 1240 points**

****

**Under 10000 points**

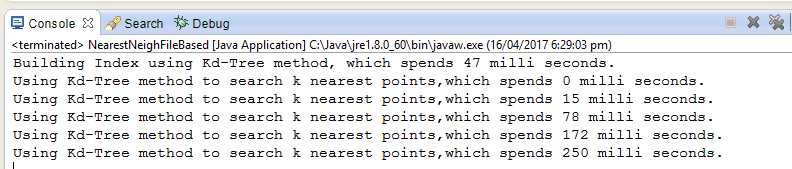
****

**Under 20000 points**

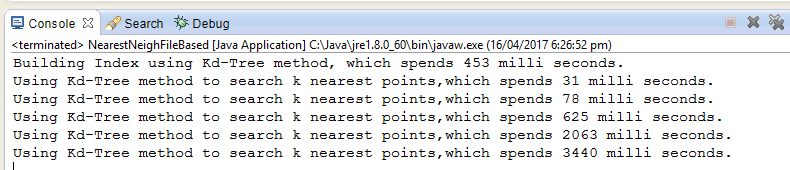
****

**Using Kd-Tree method to search 1,10,100,300,500 nearest points respectively.**

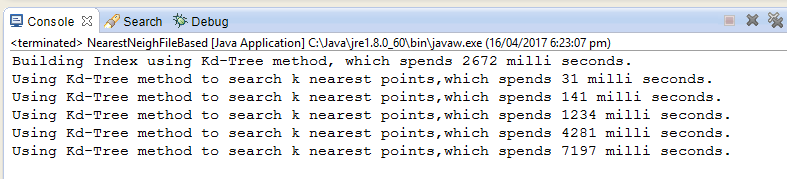
**Under 1240 points**

****

**Under 10000 points**

****

**Under 20000 points**

****

**Scenario 2 Naïve method**

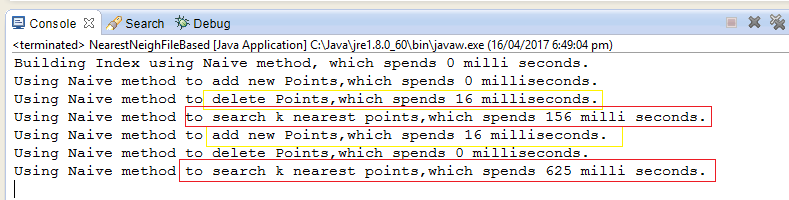
**Under 1240 points**

Using naïve method to delete 500 points ,which spends **16** mill seconds

After delete these 500 points , Search 10 nearest points using naïve method which spends **156** mill seconds

Using naïve method to add 500 same points ,which spends **16** mill seconds

After adding these 500 points , Search 10 nearest points using naïve method which spends **625** mill seconds

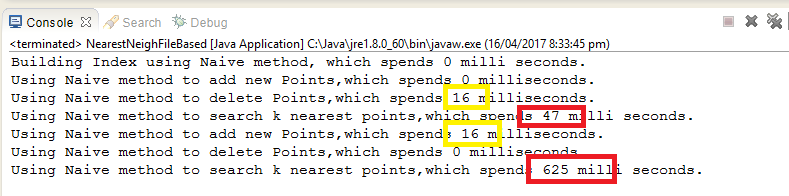


Using naïve method to delete 800 points ,which spends **16** mill seconds

After delete these 800 points , Search 10 nearest points using naïve method which spends **47** mill seconds

Using naïve method to add 800 same points ,which spends **16** mill seconds

After adding these 800 points , Search 10 nearest points using naïve method which spends **625** mill seconds



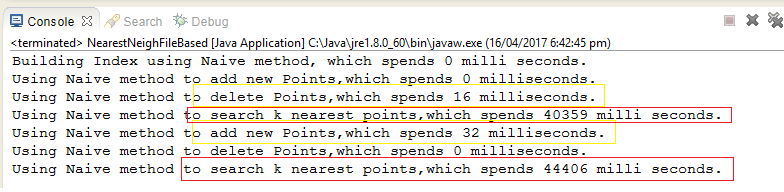
**Under 10000 points**

Using naïve method to delete 500 points ,which spends **16** mill seconds

After delete these 500 points , Search 10 nearest points using naïve method which spends **40359** mill seconds

Using naïve method to add 500 same points ,which spends **32** mill seconds

After adding these 500 points , Search 10 nearest points using naïve method which spends **44406** mill seconds

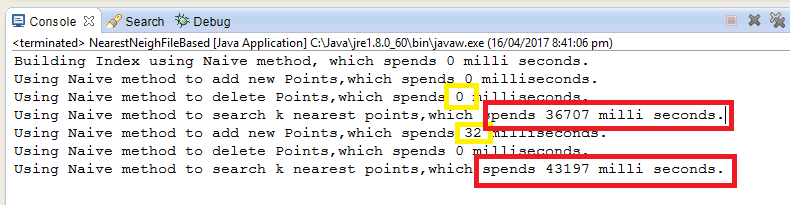


Using naïve method to delete 800 points ,which spends **0** mill seconds

After delete these 800 points , Search 10 nearest points using naïve method which spends **36707** mill seconds

Using naïve method to add 800 same points ,which spends **32** mill seconds

After adding these 800 points , Search 10 nearest points using naïve method which spends **43179** mill seconds



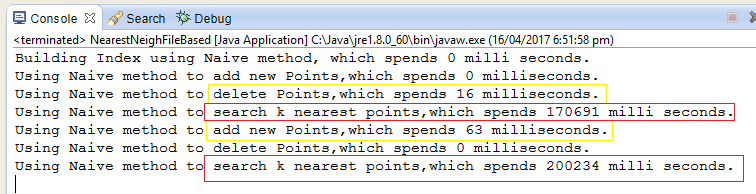
**Under 20000 points**

Using naïve method to delete 500 points , which spends **16** mill seconds

After delete these 500 points , Search 10 nearest points using naïve method which spends **170691** mill seconds

Using naïve method to add 500 same points , which spends **63** mill seconds

After adding these 500 points , Search 10 nearest points using naïve method which spends **200234** mill seconds

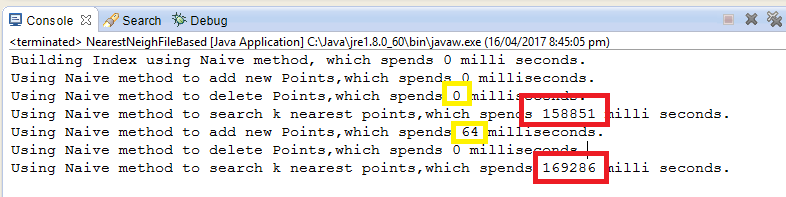


Using naïve method to delete 800 points , which spends **0** mill seconds

After delete these 800 points , Search 10 nearest points using naïve method which spends **158851** mill seconds

Using naïve method to add 800 same points , which spends **64** mill seconds

After adding these 800 points , Search 10 nearest points using naïve method which spends **169286** mill seconds

****

**Scenario 2 Using Kd-Tree method**

**Under 1240 points**

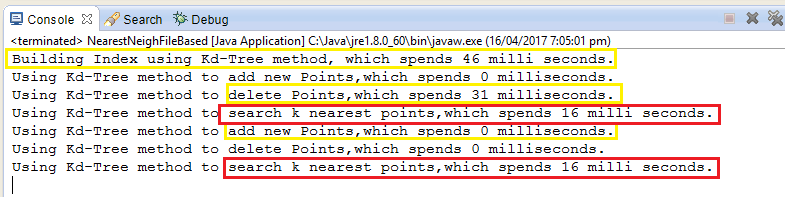
Building kd-Tree data structure spends 46 milliseconds.

Using kd-tree method to delete 500 points , which spends 31 mill seconds

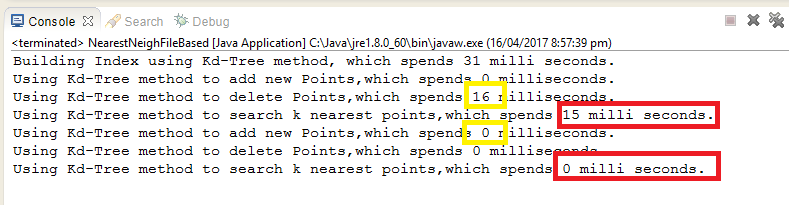
After delete these 500 points , Search 10 nearest points using kd-tree method which spends 16 mill seconds

Using kd-tree method to add 500 same points , which spends 0 mill seconds

After adding these 500 points , Search 10 nearest points using kd-tree method which spends 16 mill seconds

****

**DELETE 800 POINTS**

****

**under 10000 points**

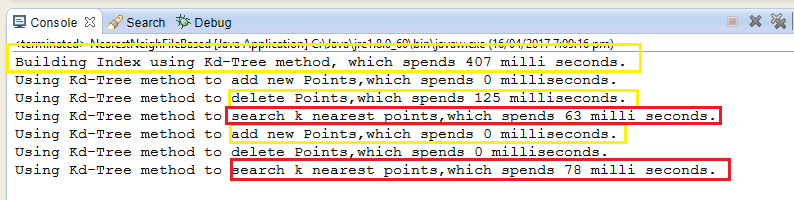
Building kd-Tree data structure spends **407** milliseconds.

Using kd-tree method to delete 500 points , which spends **125** mill seconds

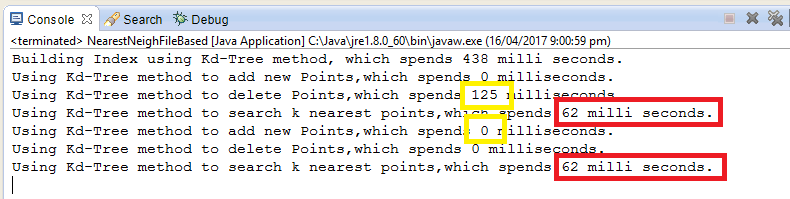
After delete these 500 points , Search 10 nearest points using kd-tree method which spends **63** mill seconds

Using kd-tree method to add 500 same points , which spends  **0** mill seconds

After adding these 500 points , Search 10 nearest points using kd-tree method which spends **78** mill seconds

****

**DELETE 800 POINTS**

****

**under 20000 points**

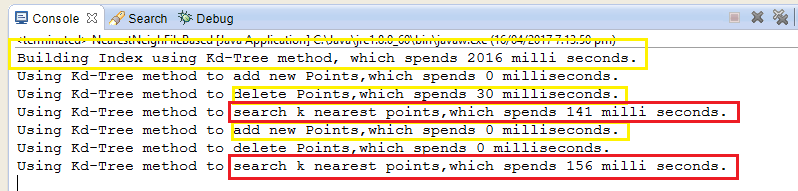
Building kd-Tree data structure spends **2016** milliseconds.

Using kd-tree method to delete 500 points , which spends **30** mill seconds

After delete these 500 points , Search 10 nearest points using kd-tree method which spends **141** mill seconds

Using kd-tree method to add 500 same points , which spends **0** mill seconds

After adding these 500 points , Search 10 nearest points using kd-tree method which spends **156** mill seconds



**DELETE 800 POINTS**

