Laporan Pemerolehan Informasi UTS 2

Pencarian menggunakan Inverted Index

Dosen Pengampu : JB. Budi Darmawan S.T., M.Sc.



Disusun oleh

Nama : Dyline Melynea Fernandez

NIM : 185314125

PROGRAM STUDI INFORMATIKA

FAKULTAS SAINS & TEKNOLOGI

UNIVERSITAS SANATA DHARMA

1. Rancangan struktur data

Untuk **index** dan **sumber** menggunakan HashMap yang prinsipnya seperti kamus/dictionary. Namun value dari HashMap **index** dibuat struktur datanya yaitu HashSet yang merupakan himpunan. Sehingga value dari key HashMap **index** bisa lebih dari 1 elemen. **index** merupakan pemetaan antara term sebagai key dan id dokumen/indeks dokumen sebagai valuenya, **sumber** merupakan pemetaan antara id dokumen/indeks dokumen sebagai key dan nama file dokumen sebagai valuenya.

Ilustrasi:

**HashMap index**

Term  
  
teknik  
semarang  
membacakan  
mempunyai  
mendobrak  
saku  
lengang  
spektakuler  
daripada  
bintung  
tersihir  
muda  
...

HashSet <Integer>

{ 2 }  
{ 0,1,3,4,5,6,7,8,9 }  
{ 11 }  
{ 9 }   
{ 3 }  
{ 11 }  
{ 11 }  
{ 11 }  
{ 11 }  
{ 10 }  
{ 7 }  
{ 9 }

...

**HashMap sumber**

Indeks dokumen

0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11

Nama File

doc1.txt  
doc10.txt  
doc11.txtdoc12.txtdoc2.txt  
doc3.txt  
doc4.txt  
doc5.txt  
doc6.txt  
doc7.txt  
doc8.txt  
doc9.txt

Konektor warna-warni hanyalah gambaran di method search, setelah didapatkan himpunan integer, maka memanggil HashMap **sumber** untuk mendapatkan nama file dokumen.

Gambaran output HashMap **sumber** dan **index** di program

1. Algoritma searching more than 1 keyword
2. Pisahkan setiap term/kata keywords (patokan pisah dengan whistespace)
3. Buat objek HashSet untuk menampung index term (tipe Integer)
4. Cari irisan index dari semua keywords (gunakan fungsi retainAll dari kelas HashSet)
5. Apabila ukuran dari objek hashset yang menampung index term adalah 0, maka cetak “tidak ditemukan”
6. Lakukan perulangan sebanyak jumlah elemen objek HashSet
7. Cetak mapping values dari key yang bersesuaian iterasi foreach objek HashSet

Penjelasan:

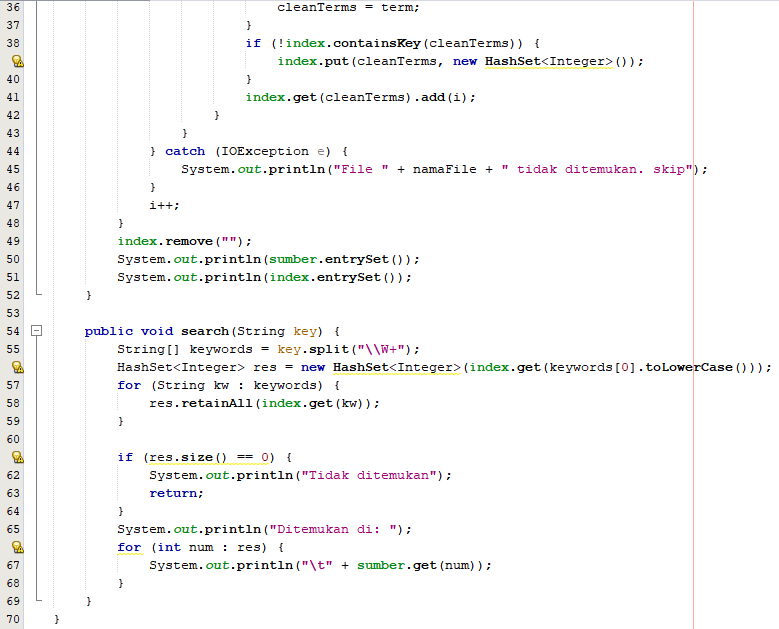
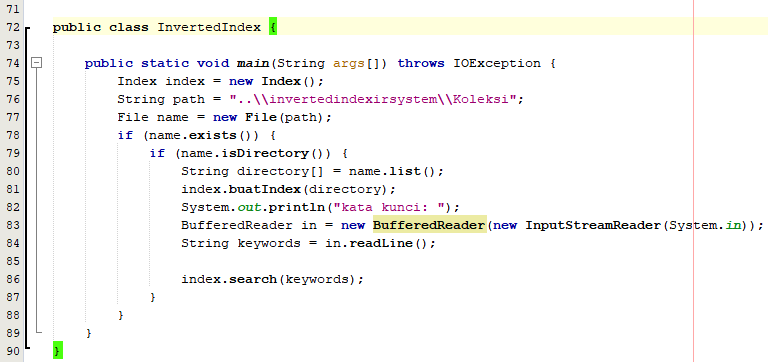
Keyword dipisah terlebih dahulu untuk setiap katanya, kemudian membuat objek HashSet untuk menampung indeks term. Untuk awal pertama membuat objek, isi parameter konstruktor HashSet diisi dengan **index.get(keywords[0].toLowerCase())** untuk inisialisasi. Setelahnya isi dari objek HashSet tersebut (**res**), akan mengalami perulangan sebanyak banyak elemen keyword yang sudah dipisahkan per kata. Pada perulangan tersebut, akan ditetapkan value **res** yang beririsan dengan hasil indeks term.

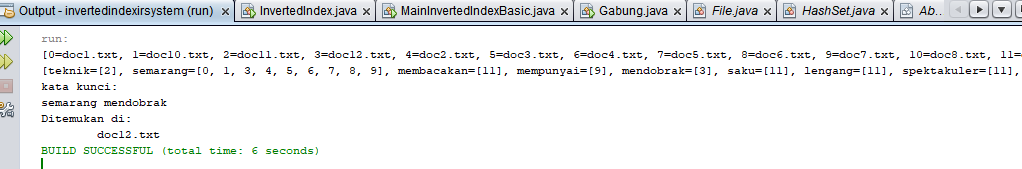
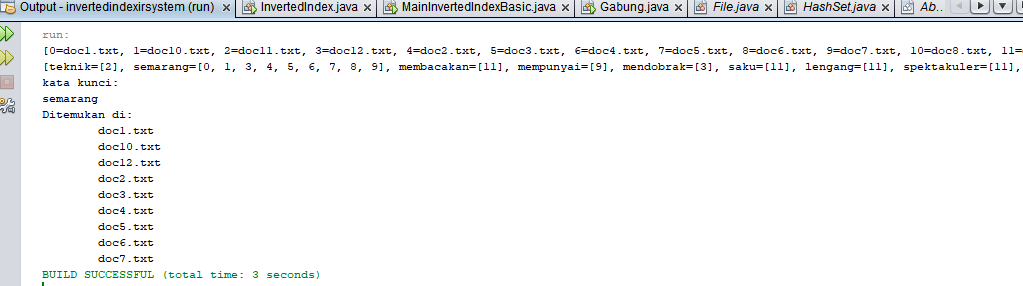
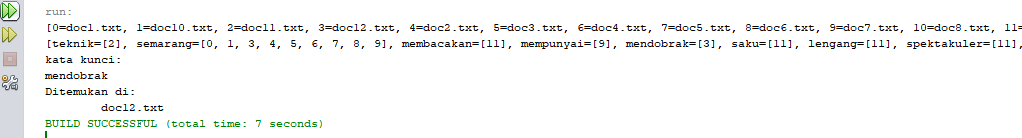
Kemudian selanjutnya memfilter kondisi jika ukuran objek **res** adalah 0, yang artinya tidak ada index term tersebut, maka cetak “tidak ditemukan”.

Setelah sudah melakukan kondisi di atas, di luar kondisi tersebut dilakukan foreach loop sebanyak elemen **res**. Dan mencetak atribut **sumber** tipe HashMap yang menampung indeks term sebagai key dengan nama filenya sebagai value. Hasil atribut itu adalah dokumen yang mewadahi keyword yang dicari.

1. Program (captured)





**Output:**