



## SESI 5

### PEMOGRAMMAN 2

#### RegexCheckData.java

#### Source code :

```
package regex;

import java.util.regex.*;

public class RegexCheckData{
    public static void main(String[] args){
        String text1 = "Matheas Jabs 50092 KA (021)-28946300 AA SA mjbs@yahoo.com";

        // mencari kata yang terdiri dari huruf A s/d Z atau a s/d z dan panjang antara 2
        // sampai 20 karakter
        cekRegex("\\w[A-Za-z]{2,20}\\w",text1);
    }

    public static void cekRegex(String regexp,String kata){
        Pattern p = Pattern.compile(regexp);
        Matcher m = p.matcher(kata);

        while(m.find()){
            if(m.group().length()!=0){
                System.out.println(m.group().trim());
            }
            System.out.println("indeks awal : "+m.start());
            System.out.println("indeks akhir : "+m.end());
        }
    }
}
```

#### Penjelasan :

1. import semua kelas yang berada pada paket java.util.regex.
2. pada method main dideklarasikan variabel text1 = "Matheas Jabs 50092 KA (021)-28946300 AA SA mjbs@yahoo.com" dengan tipe data String.
3. mengirimkan parameter "\\w[A-Za-z]{2,20}\\w" dan nilai dari variabel text1 ke method cekRegex. maksud dari baris "\\w[A-Za-z]{2,20}\\w" adalah untuk menemukan kata yang secara terurut terdiri dari satu karakter angka atau huruf yang diikuti huruf saja baik kapital ataupun kecil dengan panjang huruf dari indeks 2 sampai batas indeks 20,dan 1 huruf atau angka lagi. sehingga pada nilai dari variabel text1 yang memenuhi adalah *Matheas*, *Jabs*, *mjbs* dan *yahoo*.
4. akhir dari method main
5. pada method cekRegex() parameter diterima secara berturut-turut oleh variabel regexp dan kata.



6. Dideklarasikan objek `p= Pattern.compile(regex)` terhadap kelas `Pattern`. pada method `compile` dikirimkan ekspresi untuk mencari kata yang telah dijelaskan sebelumnya.
7. Dideklarasikan objek `m = p.matcher(kata)` terhadap kelas `Matcher`. Pada method `matcher` dikirimkan nilai dari variabel `kata`.
8. terdapat pengulangan `while` yang pernyataannya merupakan method `find()` pada kelas `Matcher`. method `find` ini akan mengembalikan nilai `true` jika terjadi kecocokan antara `pattern` dengan `kata`.
9. jika nilainya bernilai `true` maka akan menjalankan : `if(m.group().length()!=0){}` maksud dari pernyataan pada `if` adalah **jika panjang dari nilai return method `group()` tidak sama dengan 0** maka pernyataan bernilai benar.
10. jika pernyataan pada selection benar maka akan menampilkan **salinan string(mengabaikan whitespace) dari nilai return method `group()`**.
11. akhir dari selection.
12. menampilkan indeks awal dari kata yang cocok
13. menampilkan indeks batas indeks dari kata yang cocok
14. akhir dari pengulangan dan method `cekRegex`

#### Printscreen

```
C:\Users\Muhammad Sidratul M\Documents\UP\muhammad sidratul muntaha\semester 4\pe
mogranan 2\kuliah-pemograman-2\prak\sessi5\belajar-regex>mvn exec:java -Dexec.mai
nClass="regex.RegexCheckData" -Dexec.classpathScope=runtime
[INFO] Scanning for projects...
[INFO] Searching repository for plugin with prefix: 'exec'.
[INFO]
[INFO] Building belajar-regex
[INFO] task-segment: [exec:java]
[INFO]
[INFO] Preparing exec:java
[INFO] No goals needed for project - skipping
[INFO] exec:java (execution: default-cli)
Matheas
indeks awal : 0
indeks akhir : 7
Jabs
indeks awal : 8
indeks akhir : 12
mjbs
indeks awal : 43
indeks akhir : 47
yahoo
indeks awal : 48
indeks akhir : 53
[INFO]
[INFO] BUILD SUCCESSFUL
[INFO]
[INFO] Total time: 1 second
[INFO] Finished at: Sun Jun 24 22:20:13 ICT 2012
[INFO] Final Memory: 6M/15M
[INFO]
C:\Users\Muhammad Sidratul M\Documents\UP\muhammad sidratul muntaha\semester 4\pe
mogranan 2\kuliah-pemograman-2\prak\sessi5\belajar-regex>
C:\Users\Muhammad Sidratul M\Documents\UP\muhammad sidratul muntaha\semester 4\pe
```

Gambar1. Hasil dari `RegexCheckData.java`



### RegexCheckNumber.java

```
package regex;

import java.util.regex.*;

public class RegexCheckNumber{
    public static void main(String[] args){
        String text1 ="Matheas Jabs 50092 KA (021)-28946300 AA SA mjbs@yahoo.com";
        //mencari angka yang terdiri dari 5 digit misalnya untuk mencari kode post
        regexChecker("\\s\\d{5}\\s",text1);
    }

    public static void regexChecker(String regExp,String isi){
        Pattern regexP = Pattern.compile(regExp);
        Matcher regexM = regexP.matcher(isi);

        while(regexM.find()){
            if(regexM.group().length()!=0){
                System.out.println(regexM.group().trim());
            }

            System.out.println("start indeks : "+regexM.start());
            System.out.println("end indeks : "+regexM.end());
        }
    }
}
```

#### Penjelasan :

1. import semua kelas yang berada pada paket java.util.regex.
2. pada method main dideklarasikan variabel text1 = "Matheas Jabs 50092 KA (021)-28946300 AA SA mjbs@yahoo.com" dengan tipe data String.
3. mengirimkan parameter "\\s\\d{5}\\s " dan nilai dari variabel text1 ke method cekRegex. maksud dari baris "\\s\\d{5}\\s " adalah untuk menemukan kata yang secara terurut terdiri dari whitespace, 5digit angka dan whitespace lagi. sehingga pada nilai dari variabel text1 yang memenuhi adalah " 50092 " .
4. akhir dari method main
5. pada method cekRegex() parameter diterima secara berturut-turut oleh variabel regexp dan kata.
6. Dideklarasikan objek p= Pattern.compile(regexp) terhadap kelas Pattern. pada method compile dikirimkan ekspresi untuk mencari kata yang telah dijelaskan sebelumnya.
7. Dideklarasikan objek m = p.matcher(kata) terhadap kelas Matcher. Pada method matcher dikirimkan nilai dari variabel kata.
8. terdapat pengulangan while yang pernyataannya merupakan method find() pada kelas Matcher.method find ini akan mengembalikan nilai true jika terjadi kecocokan antara pattern dengan kata.



9. jika nilainya bernilai true maka akan menjalankan : `if(m.group().length()!=0){}` maksud dari pernyataan pada if adalah **jika panjang dari nilai return method group() tidak sama dengan 0** maka pernyataan bernilai benar.
10. jika pernyataan pada selection benar maka akan menampilkan **salinan string(mengabaikan whitespace)** dari nilai return method group().
11. akhir dari selection.
12. menampilkan indeks awal dari kata yang cocok
13. menampilkan indeks batas indeks dari kata yang cocok
14. akhir dari pengulangan dan method cekRegex

#### Printscreen

```
C:\Users\Muhammad Sidratul M\Documents\UP\muhammad sidratul muntaha\semester 4\pe  
mogranan 2\kuliah-pemograman-2\prak\sesi5\belajar-regex>mvn exec:java -Dexec.mai  
nClass="regex.RegexCheckNumber" -Dexec.classpathScope=runtime  
[INFO] Scanning for projects...  
[INFO] Searching repository for plugin with prefix: 'exec'.  
[INFO] -----  
[INFO] Building belajar-regex  
[INFO] task-segment: [exec:java]  
[INFO] -----  
[INFO] Preparing exec:java  
[INFO] No goals needed for project - skipping  
[INFO] [exec:java {execution: default-cli}]  
50092  
start indekx : 12  
end indekx : 19  
[INFO] -----  
[INFO] BUILD SUCCESSFUL  
[INFO] -----  
[INFO] Total time: < 1 second  
[INFO] Finished at: Sun Jun 24 22:21:46 ICT 2012  
[INFO] Final Memory: 6M/15M  
[INFO] -----  
C:\Users\Muhammad Sidratul M\Documents\UP\muhammad sidratul muntaha\semester 4\pe  
mogranan 2\kuliah-pemograman-2\prak\sesi5\belajar-regex>_
```

Gambar2. hasil dari RegexCheckNumber.java



### RegexCheckEmail.java

```
package regex;

import java.util.regex.*;

public class RegexCheckEmail{
    public static void main(String[] args){
        String text1 ="Matheas Jabs 50092 KA (021)-28946300 AA SA mjbs@yahoo.com";

        //mencari format email
        regexChecker("[A-Za-z0-9._%+-]+@[A-Za-z0-9._%+-]+\.[A-Za-z]{2,4}",text1);
    }

    public static void regexChecker(String regExp,String isi){
        Pattern regexP = Pattern.compile(regExp);
        Matcher regexM = regexP.matcher(isi);

        while(regexM.find()){
            if(regexM.group().length() != 0){
                System.out.println(regexM.group().trim());
            }

            System.out.println("start index :"+regexM.start());
            System.out.println("End index :"+regexM.end());
        }
    }
}
```

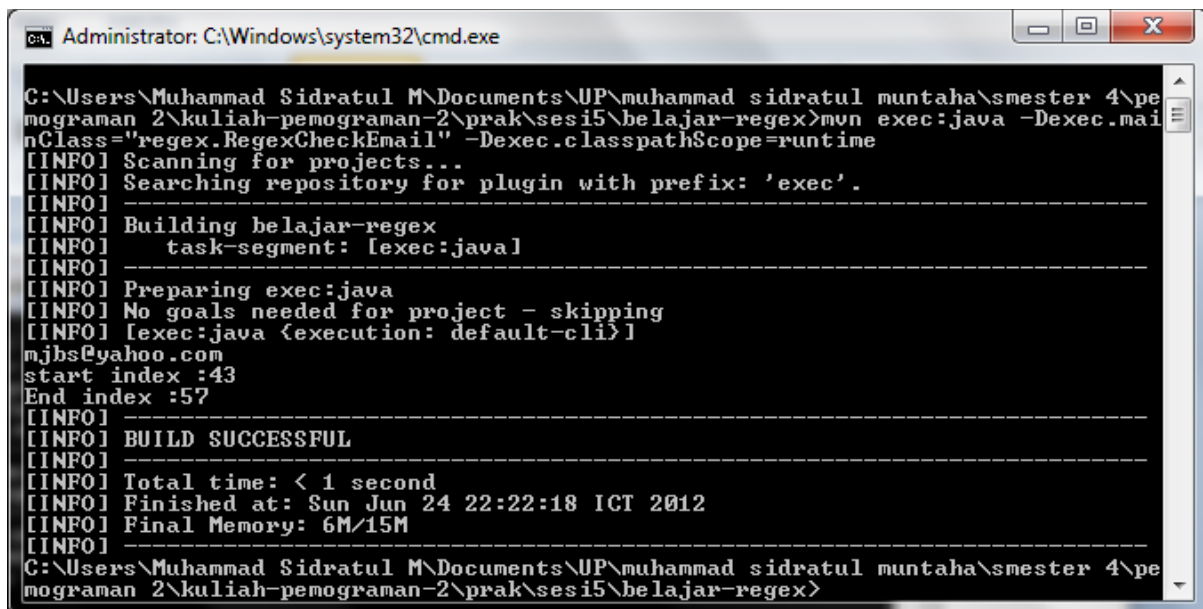
### Penjelasan

1. import semua kelas yang berada pada paket java.util.regex.
2. pada method main dideklarasikan variabel text1 = "Matheas Jabs 50092 KA (021)-28946300 AA SA mjbs@yahoo.com" dengan tipe data String.
3. mengirimkan parameter "[A-Za-z0-9.\_%+-]+@[A-Za-z0-9.\_%+-]+\.[A-Za-z]{2,4}" dan nilai dari variabel text1 ke method cekRegex. maksud dari baris "[A-Za-z0-9.\_%+-]+@[A-Za-z0-9.\_%+-]+\.[A-Za-z]{2,4}" adalah untuk menemukan kata yang secara terurut terdiri dari **1 atau lebih, huruf, angka, titik, tanda "\_", "%", "+", atau "-" ,diikuti dengan tanda "@" dan 1 atau lebih, huruf, angka, titik, tanda "\_", "%", "+", atau "-" diikuti dengan tanda "." serta 3 huruf**. sehingga pada nilai dari variabel text1 yang memenuhi adalah "mjbs@yahoo.com" .
4. akhir dari method main
5. pada method cekRegex() parameter diterima secara berturut-turut oleh variabel regexp dan kata.
6. Dideklarasikan objek p= Pattern.compile(regexp) terhadap kelas Pattern. pada method compile dikirimkan ekspresi untuk mencari kata yang telah dijelaskan sebelumnya.
7. Dideklarasikan objek m = p.matcher(kata) terhadap kelas Matcher. Pada method matcher dikirimkan nilai dari variabel kata.



8. terdapat pengulangan while yang pernyataannya merupakan method find() pada kelas Matcher.method find ini akan mengembalikan nilai true jika terjadi kecocokan antara pattern dengan kata.
9. jika nilainya bernilai true maka akan menjalankan : if(m.group().length()!=0 ){ } maksud dari pernyataan pada if adalah **jika panjang dari nilai return method group() tidak sama dengan 0** maka pernyataan bernilai benar.
10. jika pernyataan pada selection benar maka akan menampilkan **salinan string(mengabaikan whitespase) dari nilai return method group()**.
11. akhir dari selection.
12. menampilkan indeks awal dari kata yang cocok
13. menampilkan indeks batas indeks dari kata yang cocok
14. akhir dari pengulangan dan method cekRegex

Printscreen :



```
C:\Windows\system32\cmd.exe

C:\Users\Muhammad Sidratul M\Documents\UP\muhammad sidratul muntaha\smester 4\pe
mogranan 2\kuliah-pemograman-2\prak\sesi5\belajar-regex>mvn exec:java -Dexec.mai
nClass="regex.RegexCheckEmail" -Dexec.classpathScope=runtime
[INFO] Scanning for projects...
[INFO] Searching repository for plugin with prefix: 'exec'.
[INFO]
[INFO] Building belajar-regex
[INFO]   task-segment: [exec:java]
[INFO]
[INFO] Preparing exec:java
[INFO] No goals needed for project - skipping
[INFO] exec:java {execution: default-cli}
mjbs@yahoo.com
start index :43
End index :57
[INFO]
[INFO] BUILD SUCCESSFUL
[INFO]
[INFO] Total time: < 1 second
[INFO] Finished at: Sun Jun 24 22:22:18 ICT 2012
[INFO] Final Memory: 6M/15M
[INFO]
C:\Users\Muhammad Sidratul M\Documents\UP\muhammad sidratul muntaha\smester 4\pe
mogranan 2\kuliah-pemograman-2\prak\sesi5\belajar-regex>
```

Gambar 3. hasil dari RegexCheckEmail.java