

**PRAKTIKUM PEMOGRAMAN
BERORIENTASI OBYEK LANJUTAN
KELAS C**



DISUSUN OLEH :

Nama : Johanes Yogtan Wicaksono Raharja
NIM : 215314105

**TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS SANATA DHARMA YOGYAKARTA
2022**

Latihan

1. Buatlah suatu kelas Latihan StringConstructors di bawah ini!

```
4 public class StringConstructors {
5     public static void main(String[] args) {
6
7         char charArray[] = {'b', 'i', 'r', 't', 'h', ' ', 'd', 'a', 'y'};
8         String s = new String(charArray);
9
10        String s1 = new String();
11        String s2 = new String(s);
12        String s3 = new String(charArray);
13        String s4 = new String(charArray, 6, 3);
14
15        System.out.println("s1 = ");
16        System.out.println("s2 = " + s2);
17        System.out.println("s3 = " + s3);
18        System.out.println("s4 = " + s4);
19        if (s.equals("hello")) {
20            System.out.println("s is equals as \"hello\"");
21        } else {
22            System.out.println("s is not equals as \"hello\"");
23        }
24        if (s.equalsIgnoreCase("HELLO")) {
25            System.out.println("s is equals as \"hello\"");
26        } else {
27            System.out.println("s is not equals as \"hello\"");
28        }
29        if (s3.contains("day")) {
30            System.out.println("s3 contains \"day\"");
31        } else {
32            System.out.println("s3 do not contains \"day\"");
33        }
34
35        String s5 = s.concat(" " + s3);
36        System.out.println(s5);
37
38        String s6 = s5.replaceAll("hello", "happy");
39        System.out.println(s6);
40
41        String[] s7 = new String[3];
42        s7 = s6.split(" ");
43        for (int i = 0; i < s7.length; i++) {
44            System.out.println(s7[i]);
45        }
46
47        StringTokenizer tokens = new StringTokenizer(s6);
48        System.out.println("Number of Token =" + tokens.countTokens());
49
50        while (tokens.hasMoreTokens()) {
51            System.out.println(tokens.nextToken());
52        }
53    }
54 }
```

2. Buatlah suatu kelas Latihan RegexSubstitution di bawah ini!

```

10  * @author ASUS
11  */
12  public class RegexSubstitution {
13      public static void main(String args[]) {
14          String firstString = "Every day is sunday";
15          String secondString = "Vini pergi ke pasar 3 jam lalu, bersama 4 temannya";
16
17          System.out.printf("Original String 1: %s\n", firstString);
18          /*Pada baris dibawah ini berfungsi untuk menambah kata dari deklarasi
19          string pertama, dengan menggunakan sintak replaceAll dari deklarasi string
20          dilanjutkan penggunaan slash yang berfungsi untuk memberi pertambahan
21          kalimat setelah Every*/
22          firstString = firstString.replaceAll("\\bday\\b", "week there");
23          System.out.printf("Original string 1 become : %s\n", firstString);
24
25          System.out.printf("Original String 2: %s\n", secondString);
26          /*Pada baris dibawah ini berfungsi untuk mengganti huruf a dalam
27          kalimat menjadi huruf o, dengan menggunakan sintak replaceAll dari
28          deklarasi string kedua dilanjutkan penggunaan model [a] yang akan digantikan
29          huruf o*/
30          secondString = secondString.replaceAll("[a]", "o");
31          System.out.printf("Original string 2 become : %s\n", secondString);
32
33          /*Pada baris dibawah ini berfungsi untuk mengganti setiap kalimat di
34          string pertama dengan kalimat word, dengan menggunakan sintak replaceAll
35          dari deklarasi string pertama, dilanjutkan dengan model word yang
36          akan digantikan di setiap kalimat string pertama*/
37          System.out.printf("Every word replaced by \"word\": %s\n\n",
38                          firstString.replaceAll("\\w+", "word"));
39
40          /*Pada baris dibawah ini tidak bisa dicetak, karena \\d memilih angka digit
41          bukan kalimat "digit" tidak berbentuk angka*/
42          secondString = secondString.replaceFirst("\\d", "digit");
43
44
45          System.out.printf("?? %s\n", secondString); //isilah ?? dengan kalimat yang sesuai
46          String output = "String split at commas: [";
47
48          String[] results = secondString.split(",\\s*"); // jelaskan
49          /*Pada baris dibawah ini akan menjalankan sebuah perulangan yang dimana
50          sebelumnya sebuah sintak disimpan ke result dan kalimat disimpan ke variabel
51          di output yang dimana akan digabungkan dalam sebuah perulangan per variabelnya*/
52          for (String string : results) // jelaskan sintaks ini
53          {
54              output += "\"" + string + "\", ";
55          }
56          System.out.println("Hasil split : " + output);
57
58          output = output.substring(0, output.length() - 2) + "];";
59
60          System.out.println(output);
61      } // end main
62  } // end class RegexSubstitutio

```

3. Pada kasus perpustakaan tambahkan perintah berikut ini

```

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    int selectedRow = jTable1.getSelectedRow();
    tfId.setText(tabelModel.getValueAt(selectedRow, 0).toString());
    tfNo.setText(tabelModel.getValueAt(selectedRow, 1).toString());
    tfNama.setText(tabelModel.getValueAt(selectedRow, 2).toString());
    tfAlamat.setText(tabelModel.getValueAt(selectedRow, 3).toString());
    tfJum.setText(tabelModel.getValueAt(selectedRow, 4).toString());
}

private void btHapusActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int a = jTable1.getSelectedRow();
    if (a == -1) {
    } else {
        dtPeminjam.peminjamList.remove(a);
    }

    tampilData();
}

```

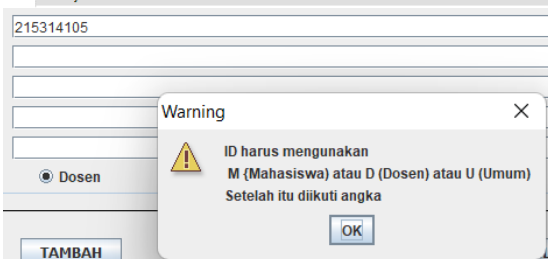
4. Tambahkan pula Exception Handling dengan menggunakan manipulasi String (regular expression) untuk mengecek input pada JTextField :

- idPeminjam harus diawali M/D/U

```

private void tfIdFocusLost(java.awt.event.FocusEvent evt) {
    // TODO add your handling code here:
    if (tfId.getText().isEmpty()) {
        return;
    }
    String cek = tfId.getText();
    if (!cek.matches("^[M|D|U][\\d+].*")) {
        try {
            throw new Exception("ID harus menggunakan\n M (Mahasiswa) atau D (Dosen) atau U (Umum) \nSetelah itu diikuti angka");
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(this, ex.getMessage(), "Warning", JOptionPane.WARNING_MESSAGE);
        }
    }
}

```

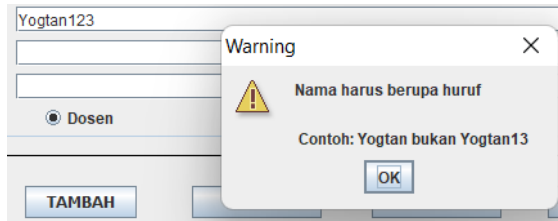


- nama tidak mengandung angka

```

private void tfNamaFocusLost(java.awt.event.FocusEvent evt) {
    // TODO add your handling code here:
    if (tfNama.getText().isEmpty()) {
        return;
    }
    String cek = tfNama.getText();
    if (cek.matches(".*\\d.*")) {
        try {
            throw new Exception("Nama harus berupa huruf \n \n Contoh: Yogtan bukan Yogtan13");
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(this, ex.getMessage(), "Warning", JOptionPane.WARNING_MESSAGE);
        }
    }
}

```



- nim berupa 9 digit angka, nik berupa 16 digit angka, nip berupa 18 digit angka

```
private void tfNoFocusLost(java.awt.event.FocusEvent evt) {
    // TODO add your handling code here:
    if (tfNo.getText().isEmpty()) {
        return;
    }
    String cek = tfNo.getText();
    if (!cek.matches("[0-9]+")) {
        try {
            throw new Exception("Nomor ID harus berupa angka \n\n Contoh "
                + ": 215314105");
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(this, ex.getMessage(),
                "Warning", JOptionPane.WARNING_MESSAGE);
        }
    }
    if (rbMhs.isSelected()) {
        if (cek.length() != 9) {
            try {
                throw new Exception("Angka harus berjumlah 9");
            } catch (Exception ex) {
                JOptionPane.showMessageDialog(this, ex.getMessage(),
                    "Warning", JOptionPane.WARNING_MESSAGE);
                return;
            }
        }
    }
    if (rbDosen.isSelected()) {
        if (cek.length() != 16) {
            try {
                throw new Exception("Angka harus berjumlah 16");
            } catch (Exception ex) {
                JOptionPane.showMessageDialog(this, ex.getMessage(),
                    "Warning", JOptionPane.WARNING_MESSAGE);
                return;
            }
        }
    }
    if (rbUmum.isSelected()) {
        if (cek.length() != 18) {
            try {
                throw new Exception("Angka harus berjumlah 18");
            } catch (Exception ex) {
                JOptionPane.showMessageDialog(this, ex.getMessage(),
                    "Warning", JOptionPane.WARNING_MESSAGE);
                return;
            }
        }
    }
}
}
```

NIM	12345678
Nama	
Alamat	
Jumlah Pinjam	
Jenis Peminjam	<input type="radio"/> Dosen

Warning

Angka harus berjumlah 9

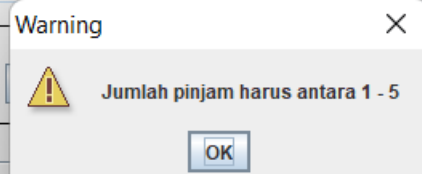
- maks pinjam untuk Mahasiswa : 5, Dosen : 7, Umum : 3

```
private void tfJumFocusLost(java.awt.event.FocusEvent evt) {
    // TODO add your handling code here:
    if (tfJum.getText().isEmpty()) {
        return;
    }
    String cek = tfJum.getText();
    if (!cek.matches("[0-9]+")) {
        try {
            throw new Exception("Jumlah Pinjam harus berupa angka");
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(this, ex.getMessage(), "Warning", JOptionPane.WARNING_MESSAGE);
            return;
        }
    }
    if (rbMhs.isSelected()) {
        if (Integer.parseInt(cek) > 5) {
            try {
                throw new Exception("Jumlah pinjam harus antara 1 - 5");
            } catch (Exception ex) {
                JOptionPane.showMessageDialog(this, ex.getMessage(), "Warning", JOptionPane.WARNING_MESSAGE);
                return;
            }
        }
    }
    if (rbDosen.isSelected()) {
        if (Integer.parseInt(cek) > 7) {
            try {
                throw new Exception("Jumlah pinjam harus antara 1 - 7");
            } catch (Exception ex) {
                JOptionPane.showMessageDialog(this, ex.getMessage(), "Warning", JOptionPane.WARNING_MESSAGE);
                return;
            }
        }
    }
    if (rbUmum.isSelected()) {
        if (Integer.parseInt(cek) > 3) {
            try {
                throw new Exception("Jumlah pinjam harus antara 1 - 3");
            } catch (Exception ex) {
                JOptionPane.showMessageDialog(this, ex.getMessage(), "Warning", JOptionPane.WARNING_MESSAGE);
                return;
            }
        }
    }
}
}
```

Jumlah Pinjam

Jenis Peminjam ☐ Dosen ☒ Mahasiswa

Id Peminjam



- idKoleksi harus diawali D/B/M

```
private void tfIdKFocusLost(java.awt.event.FocusEvent evt) {
    // TODO add your handling code here:
    if (tfIdK.getText().isEmpty()) {
        return;
    }
    String cek = tfIdK.getText();
    if (!cek.matches("[M|D|B][\\d+].*")) {
        try {
            throw new Exception("ID harus diawali dengan huruf \n M (untuk Majalah) atau D (untuk Disk) atau B (untuk Buku) \n "
                + "dan kemudian diikuti dengan angka \n \n Contoh : M1234 atau D1234 atau B1234");
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(this, ex.getMessage(), "Warning", JOptionPane.WARNING_MESSAGE);
        }
    }
}
}
```

Id Koleksi

Tahun Terbit

Judul

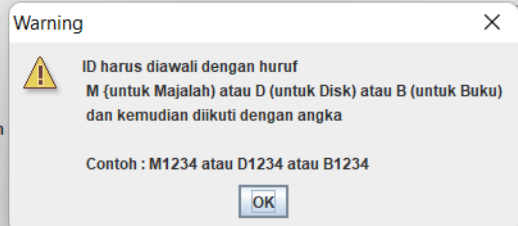
Penerbit

Status Pinjam

Volume

Seri

ISSN



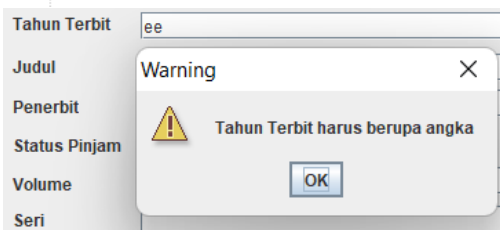
- tahunTerbit berupa 4 digit angka tahun

```

private void tfTTFocusLost(java.awt.event.FocusEvent evt) {
    // TODO add your handling code here:
    if (tfTT.getText().isEmpty()) {
        return;
    }
    String cek = tfTT.getText();
    if (!cek.matches("[0-9]+")) {
        try {
            throw new Exception("Tahun Terbit harus berupa angka");
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(this, ex.getMessage(),
                "Warning", JOptionPane.WARNING_MESSAGE);
            return;
        }
    }

    if (cek.length() != 4) {
        try {
            throw new Exception("Tahun terbit harus 4 angka");
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(this, ex.getMessage(),
                "Warning", JOptionPane.WARNING_MESSAGE);
            return;
        }
    }
}

```



- volume, seri, halaman berupa digit integer

```

private void tfVolumeFocusLost(java.awt.event.FocusEvent evt) {
    // TODO add your handling code here:
    if (tfVolume.getText().isEmpty()) {
        return;
    }
    String cek = tfVolume.getText();
    if (rbMajalah.isSelected()) {
        if (!cek.matches("[0-9]+")) {
            try {
                throw new Exception("Volume harus berupa angka");
            } catch (Exception ex) {
                JOptionPane.showMessageDialog(this, ex.getMessage(),
                    "Warning", JOptionPane.WARNING_MESSAGE);
                return;
            }
        }
    }
}

```




- format Disk berupa “Audio”, “Video” atau “Document”

```

        if (rbDisk.isSelected()) { //Perbaiki
            if (!cek.matches("^([Audio|Video|Document][\\D].*)")) {
                try {
                    throw new Exception("Format harus berupa Audio, Video, Document");
                } catch (Exception ex) {
                    JOptionPane.showMessageDialog(this, ex.getMessage(),
                        "Warning", JOptionPane.WARNING_MESSAGE);
                    return;
                }
            }
        }
    }
}

```

