

KAWAH EDUKASI

BACKEND BATCH VI

Tugas : Test Minggu Ke-4 (Regex, List & Map)

Nama : Isep Lutpi Nur

Email : iseplutpinur7@gmail.com

1. Regular Expression

a. Email

- **Regular Expression:**

`/^[a-zA-Z0-9._]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}/gm`

- **Penjelasan:**

`^[a-zA-Z0-9._]`: Menerima semua huruf kecil dan besar kemudian angka serta titik dan garis bawah dan harus di awal dari baris.

`@[a-zA-Z0-9.-]`: Harus ada simbol at (@) kemudian Menerima semua huruf kecil dan besar kemudian angka serta titik dan garis bawah.

`\.[a-zA-Z]{2,}`: Harus ada simbol titik kemudian menerima huruf kecil dan besar minimal 2 huruf.

- **Contoh Teks Yang Dapat Diterima:**

Iseplutpinur7@gmail.com

Isep.lutpi@gmail.com

upi@github.co

bambang_s@m.co

- **Contoh Teks Yang Tidak Dapat Diterima:**

salah+tes@mail.co

email+@mail.com

email@mail.c

b. Pattern Tanggal : dd-MM-yyyy

- **Regular Expression:**

`/(0[1-9]|[12][0-9]|3[01])-(0[1-9]|1[0-2])-(\d{4})/gm`

- **Penjelasan:**

`(0[1-9]|[12][0-9]|3[01])`: Group Menerima angka dari 0 sampai 9 tapi harus ada 0 sebelumnya **atau** angka 1 atau 2 yang di ikuti angka 0 sampai 9 **atau** angka 3 yang ujungnya 0 atau 1

`(0[1-9]|1[0-2])`: Group Menerima angka dari 0 sampai 9 tapi harus ada 0 sebelumnya **atau** angka 1 yang ujungnya 0 sampai 2

`(\d{4})`: Group yang menerima angka dengan 4 digit

- **Contoh Teks Yang Dapat Diterima:**

10-08-2000

01-11-2000

01-11-2022

- **Contoh Teks Yang Tidak Dapat Diterima:**

10-08-20000

1-08-20000

1-8-20000

c. **Pattern Tanggal : dddd, MMMM yy**

- **Regular Expression:**

`/([a-zA-Z]{3,}), ([a-zA-Z]{3,}) (\d{2})$/gm`

- **Penjelasan:**

`([a-zA-Z]{3,})`: Group yang menerima semua huruf minimal 3 digit

`([a-zA-Z]{3,})` : Group yang menerima semua huruf minimal 3 digit

`(\d{2})` Group yang menerima 2 digit angka

- **Contoh Teks Yang Dapat Diterima:**

Selasa, Juni 20

Sel, Jun 20

kam, november 21

- **Contoh Teks Yang Tidak Dapat Diterima:**

kamis, 20 november 2021

k, nov 20

kam, no 20

d. **Nomor Telephone Indonesia**

- **Regular Expression:**

`\+62\s?[0-9]{3,4}\s?[0-9]{3,4}\s?[0-9]{3,4}$/gm`

- **Penjelasan:**

`\+62` : Harus ada +62

`\s?` : Spasi opsional

`[0-9]{3,4}`: Menerima angka 3 sampai 4 digit

`\s?` : Spasi opsional

`[0-9]{3,4}`: Menerima angka 3 sampai 4 digit

`\s?` : Spasi opsional

`[0-9]{3,4}`: Menerima angka 3 sampai 4 digit

`$` : Mencocokkan awalan dan akhiran baris

- **Contoh Teks Yang Dapat Diterima:**

+6285798132505

+62 857 789 888

+62 857 9813 2505

- **Contoh Teks Yang Tidak Dapat Diterima:**

+628579813250555

+62 8579 81311 2505

+62 8579 81 2505

2. List

a. List Nama makanan minimal 10 data.

```
List<String> makananList = new ArrayList<>(List.of("Gorengan", "Batagor",  
    "Cilok", "Baso",  
    "Mie Ayam", "Kebab",  
    "Dimsum", "Seblak",  
    "Basreng", "Martabak"));  
System.out.println("List Makanan:");  
System.out.println(makananList);  
// output: [Gorengan, Batagor, Cilok, Baso, Mie Ayam, Kebab, Dimsum, Seblak,  
Basreng, Martabak]
```

b. List Tahun Piala Dunia minimal 3 data.

```
List<Integer> pialaDuniaList = new ArrayList<>();  
pialaDuniaList.add(2010);  
pialaDuniaList.add(2014);  
pialaDuniaList.add(2018);  
pialaDuniaList.add(2022);  
System.out.println("\nList Piala Dunia:");  
System.out.println(pialaDuniaList);  
// output: [2010, 2014, 2018, 2022]
```

c. Implement sort pada list nama provinsi minimal 10 data secara descending.

```
List<String> provinsiList = new ArrayList<>(List.of("ACEH", "SUMATERA UTARA",  
    "SUMATERA BARAT", "RIAU", "JAMBI", "SUMATERA SELATAN", "BENGKULU", "LAMPUNG",  
    "KEPULAUAN BANGKA BELITUNG", "KEPULAUAN RIAU", "DKI JAKARTA", "JAWA BARAT", "JAWA  
TENGAH", "DI YOGYAKARTA", "JAWA TIMUR", "BANTEN", "BALI", "NUSA TENGGARA BARAT",  
    "NUSA TENGGARA TIMUR", "KALIMANTAN BARAT", "KALIMANTAN TENGAH", "KALIMANTAN  
SELATAN", "KALIMANTAN TIMUR", "KALIMANTAN UTARA", "SULAWESI UTARA", "SULAWESI  
TENGAH", "SULAWESI SELATAN", "SULAWESI TENGGARA", "GORONTALO", "SULAWESI BARAT",  
    "MALUKU", "MALUKU UTARA", "PAPUA BARAT", "PAPUA"));  
System.out.println("\nList Data Provinsi:");  
System.out.println(provinsiList);  
// output: [ACEH, SUMATERA UTARA, SUMATERA BARAT, RIAU, JAMBI, SUMATERA SELATAN,  
BENGKULU, LAMPUNG, KEPULAUAN BANGKA BELITUNG, KEPULAUAN RIAU, DKI JAKARTA, JAWA  
BARAT, JAWA TENGAH, DI YOGYAKARTA, JAWA TIMUR, BANTEN, BALI, NUSA TENGGARA BARAT,  
NUSA TENGGARA TIMUR, KALIMANTAN BARAT, KALIMANTAN TENGAH, KALIMANTAN SELATAN,  
KALIMANTAN TIMUR, KALIMANTAN UTARA, SULAWESI UTARA, SULAWESI TENGAH, SULAWESI  
SELATAN, SULAWESI TENGGARA, GORONTALO, SULAWESI BARAT, MALUKU, MALUKU UTARA,  
PAPUA BARAT, PAPUA]  
  
System.out.println("\nList Data Provinsi Sort Descending:");  
provinsiList.sort(Comparator.reverseOrder());  
System.out.println(provinsiList);  
// output: [SUMATERA UTARA, SUMATERA SELATAN, SUMATERA BARAT, SULAWESI UTARA,  
SULAWESI TENGGARA, SULAWESI TENGAH, SULAWESI SELATAN, SULAWESI BARAT, RIAU, PAPUA]
```

```
BARAT, PAPUA, NUSA TENGGARA TIMUR, NUSA TENGGARA BARAT, MALUKU UTARA, MALUKU, LAMPUNG, KEPULAUAN RIAU, KEPULAUAN BANGKA BELITUNG, KALIMANTAN UTARA, KALIMANTAN TIMUR, KALIMANTAN TENGAH, KALIMANTAN SELATAN, KALIMANTAN BARAT, JAWA TIMUR, JAWA TENGAH, JAWA BARAT, JAMBI, GORONTALO, DKI JAKARTA, DI YOGYAKARTA, BENGKULU, BANTEN, BALI, ACEH]
```

d. Implement penghapusan data pada sebuah list.

```
System.out.println("\nList Data Provinsi Hapus data Indeks ke 0:");
provinsiList.remove(0);
System.out.println(provinsiList);
// output: [SUMATERA SELATAN, SUMATERA BARAT, SULAWESI UTARA, SULAWESI TENGGARA,
SULAWESI TENGAH, SULAWESI SELATAN, SULAWESI BARAT, RIAU, PAPUA BARAT, PAPUA, NUSA
TENGGARA TIMUR, NUSA TENGGARA BARAT, MALUKU UTARA, MALUKU, LAMPUNG, KEPULAUAN
RIAU, KEPULAUAN BANGKA BELITUNG, KALIMANTAN UTARA, KALIMANTAN TIMUR, KALIMANTAN
TENGAH, KALIMANTAN SELATAN, KALIMANTAN BARAT, JAWA TIMUR, JAWA TENGAH, JAWA
BARAT, JAMBI, GORONTALO, DKI JAKARTA, DI YOGYAKARTA, BENGKULU, BANTEN, BALI,
ACEH]
```

3. Map

a. employees

```
{"employees": [
  {"name": "Shyam", "email": "shyamjaiswal@gmail.com"},
  {"name": "Bob", "email": "bob32@gmail.com"},
  {"name": "Jai", "email": "jai87@gmail.com"}
]}
```

```
Map<String, Object> employees = new HashMap();

employees.put("employees", (new ArrayList<> (List.of(
    (new HashMap<String, String>() {{
        put("name", "Shyam");
        put("email", "shyamjaiswal@gmail.com");
    }}),
    (new HashMap<String, String>() {{
        put("name", "Bob");
        put("email", "bob32@gmail.com");
    }}),
    (new HashMap<String, String>() {{
        put("name", "Jai");
        put("email", "jai87@gmail.com");
    }}})));

System.out.println(employees);
// output: {employees=[{name=Shyam, email=shyamjaiswal@gmail.com}, {name=Bob,
email=bob32@gmail.com}, {name=Jai, email=jai87@gmail.com}]}
```

b. menu

```
{"menu": {
  "id": "file",
  "value": "File",
  "popup": {
    "menuitem": [
      {"value": "New", "onclick": "CreateDoc()"},
      {"value": "Open", "onclick": "OpenDoc()"},
      {"value": "Save", "onclick": "SaveDoc()"}
    ]
  }
}
```

```

    ]
  }
}}

```

```

Map<String, Object> menuData = new HashMap<>();
menuData.put("id", "file");
menuData.put("value", "File");

Map<String, Object> menuItem = new HashMap<>();
menuItem.put("menuitem", (new ArrayList<>(List.of(
    (new HashMap<String, String>(){{
        put("value", "New");
        put("onclick", "CreateDoc()");
    }}),
    (new HashMap<String, String>(){{
        put("value", "Open");
        put("onclick", "OpenDoc()");
    }}),
    (new HashMap<String, String>(){{
        put("value", "Save");
        put("onclick", "SaveDoc()");
    }}}
)))));

menuData.put("popup", menuItem);

Map<String, Object> menu = new HashMap<>();
menu.put("menu", menuData);
System.out.println(menu);
// output: {menu={popup={menuitem=[{onclick=CreateDoc(), value=New},
{onclick=OpenDoc(), value=Open}, {onclick=SaveDoc(), value=Save}]}, id=file,
value=File}}

```

c. widget

```

{
  "widget": {
    "debug": "on",
    "window": {
      "title": "Sample Konfabulator Widget",
      "name": "main_window",
      "width": 500,
      "height": 500
    },
    "image": {
      "src": "Images/Sun.png",
      "name": "sun1",
      "hOffset": 250,
      "vOffset": 250,
      "alignment": "center"
    },
    "text": {
      "data": "Click Here",
      "size": 36,
      "style": "bold",

```

```

        "name": "text1",
        "hOffset": 250,
        "vOffset": 100,
        "alignment": "center",
        "onMouseUp": "sun1.opacity = (sun1.opacity / 100) * 90;"
    }
}
}

```

```

Map<String, Object> widgetData = new HashMap<>();
widgetData.put("debug", "on");

// widget window
Map<String, Object> widgetWindow = new HashMap<>();
widgetWindow.put("title", "Sample Konfabulator Widget");
widgetWindow.put("name", "main_window");
widgetWindow.put("width", 500);
widgetWindow.put("height", 500);
widgetData.put("window", widgetWindow);

// widget image
Map<String, Object> widgetImage = new HashMap<>();
widgetImage.put("src", "Images/Sun.png");
widgetImage.put("name", "sun1");
widgetImage.put("hOffset", 250);
widgetImage.put("vOffset", 250);
widgetImage.put("alignment", "center");
widgetData.put("image", widgetImage);

// widget text
Map<String, Object> widgetText = new HashMap<>();
widgetText.put("data", "Click Here");
widgetText.put("size", 36);
widgetText.put("style", "bold");
widgetText.put("name", "text1");
widgetText.put("hOffset", 250);
widgetText.put("vOffset", 100);
widgetText.put("alignment", "center");
widgetText.put("onMouseUp", "sun1.opacity = (sun1.opacity / 100) * 90;");
widgetData.put("text", widgetText);

Map<String, Object> widget = new HashMap<>();
widget.put("widget", widgetData);
System.out.println(widget);
// output: {widget={image={vOffset=250, src=Images/Sun.png, name=sun1,
alignment=center, hOffset=250}, debug=on, window={name=main_window,
width=500, title=Sample Konfabulator Widget, height=500}, text={vOffset=100,
data=Click Here, size=36, name=text1, style=bold, alignment=center,
onMouseUp=sun1.opacity = (sun1.opacity / 100) * 90;;, hOffset=250}}}

```