

MODUL PRAKTIKUM PEMROSESAN BAHASA ALAMI (PENGENALAN REGULAR EXPRESSION)

Bambang Pilu Hartato, S.Kom., M.Eng.
FAKULTAS ILMU KOMPUTER | INFORMATIKA

A. Tujuan Praktikum

Praktikum ini bertujuan untuk memperkenalkan Regular Expression. Diharapkan setelah mengikuti praktikum kali ini mahasiswa dapat mendeskripsikan Regular Expression dan memanfaatkannya untuk membantu proses pengolahan teks.

B. Pembahasan

Regular Expression adalah suatu bahasa formal untuk menentukan suatu string teks. Regular Expression juga dapat didefinisikan sebagai serangkaian karakter (alfanumerik & simbol) yang mendefinisikan sebuah pola pencarian suatu string teks. Pola tersebut biasanya digunakan oleh algoritma pencarian string untuk melakukan operasi “search” atau “search and replace” pada string. Selain itu, pola tersebut juga dapat digunakan untuk memeriksa string masukan.

Terdapat beberapa aturan penulisan Regular Expression yang telah disepakati bersama. Berikut beberapa aturan tersebut.

1. Metacharacters

No	Karakter	Deskripsi	Contoh	Keterangan
1	[]	Himpunan karakter	[A-D]	Melakukan pencarian terhadap string yang mengandung huruf A sampai dengan D (A,B,C,D).
2	\	Tanda untuk escape secuance sepecial character	\.	Digunakan untuk mencari string yang mengandung karakter tertentu di mana karakter tersebut sebenarnya sudah digunakan sebagai simbol RegEx.
3	.	Semua karakter (kecuali ganti bari)	H..o	Akan mencari string yang memiliki pola H..o, sehingga yang ditemukan “Helo”, “Hexo”, “H2Oo”
4	^	Dimulai dengan	^Ekspresi	Akan mencari suatu string yang diawali

No	Karakter	Deskripsi	Contoh	Keterangan
				dengan kata “Ekspresi”
5	&	Diakhiri dengan	formal\$	Mencari string yang diakhiri dengan kata “formal”
6	*	Nol atau lebih kemunculan dari karakter yang berada sebelum tanda *.	po*a	Yang sesuai dengan hasil pencarian dengan pola tersebut adalah “poa”, “pola”, “polla”, “pollla”
7	+	Satu atau lebih kemunculan dari karakter yang berada sebelum tanda +	po+a	Yang sesuai dengan hasil pencarian dengan pola tersebut adalah “poa”, “pola”, “polla”, “pollla”
8	{}	Hampir sama dengan tanda * dan +, dengan tanda {} kita dapat menentukan jumlah tertentu kemunculan	big{2}	Hasil yang sesuai dengan pencarian tersebut adalah “biggest” namun “big” tidak akan muncul dari hasil pencarian tersebut.
9		Atau	saya daku	Melakukan pencarian terhadap string yang mengandung kata “saya” atau “daku”.

2. Special Sequences

Special sequence adalah serangkaian karakter yang diawali dengan backslash (/) dan diikuti dengan karakter-karakter tertentu.

No	Karakter	Deskripsi	Contoh
1	\A	Mencari string yang diawali dengan kata tertentu	\AThe
2	\b	Mencari string yang diawali atau diakhiri dengan kata tertentu	\bain atau ain\b

No	Karakter	Deskripsi	Contoh
3	\B	Mencari string yang tidak diawali atau tidak diakhiri dengan kata tertentu	\Bain atau ain\b
4	\d	Mencari string yang mengandung digit angka (0 – 9)	\d
5	\D	Mencari string yang tidak mengandung digit angka (0 – 9)	\D
6	\s	Mencari string yang mengandung space (white space)	\s
7	\S	Mencari string yang tidak mengandung space (white space)	\S
8	\w	Mencari string yang mengandung karakter alfabet (A-Z, a-z), angka (0 - 9), dan underscore (_)	\w
9	\W	Mencari string yang tidak mengandung karakter alfabet (A-Z, a-z), angka (0 - 9), dan underscore (_)	\W
10	\Z	Mencari string yang diakhiri dengan kata tertentu	Spain\Z

3. Sets

Sets adalah serangkaian karakter yang melambangkan himpunan karakter. Sets ditandai dengan serangkaian karakter yang berada di dalam kurung siku [].

No.	Contoh Karakter	Deskripsi
1	[abc]	Mencari string yang mengandung karakter a , b , atau c
2	[Aa]	Mencari string yang mengandung huruf A atau a
3	[a-d]	Mencari string yang mengandung huruf a sampai dengan d
4	[^abc]	Mencari string yang tidak mengandung huruf a , b , atau c
5	[0123]	Mencari string yang mengandung angka 0 , 1 , 2 , atau 3
6	[0-9]	Mencari string yang mengandung angka 0 sampai dengan 9
7	[0-5][0-9]	Mencari string yang terdiri dari 2 digit angka dengan nilai digit pertama adalah 0 sampai dengan 5 sedangkan nilai digit kedua adalah 0 sampai dengan 9
8	[a-zA-Z]	Mencari string yang mengandung huruf alfabet baik dalam bentuk lowercase ataupun uppercase.

Untuk memudahkan pemahaman Anda, silahkan Anda lakukan perintah-perintah berikut:

1. Akses Regexpal

Regexpal adalah suatu situs yang dapat kita gunakan untuk melakukan simulasi penggunaan Regex. Silahkan Anda akses Regexpal melalui tautan berikut ini <https://www.regexpal.com/>

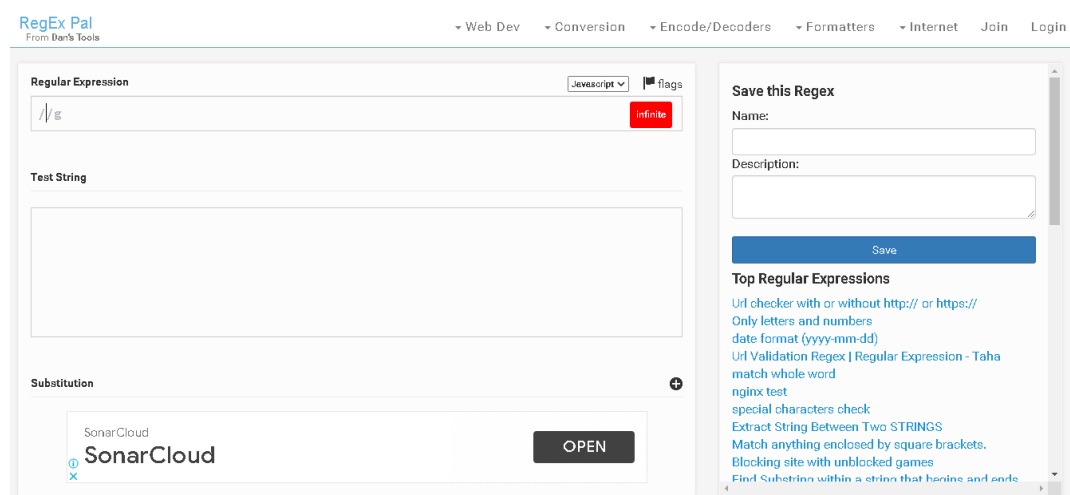


Figure 1. Tampilan Regexpal

Figure 1 adalah tampilan dari Regexpal. Kotak isian “Regular Expression” digunakan sebagai tempat bagi Anda untuk memasukkan query RegEx. Sedangkan kotak isian “Text String” digunakan sebagai tempat bagi Anda untuk menginputkan contoh string yang ingin Anda periksa RegEx-nya.

2. Cari Sampel Teks

Carilah sampel teks yang nantinya Anda inputkan ke dalam “Text String” pada Regexpal. Sebenarnya Anda dapat menggunakan contoh apa saja, namun pada contoh kali ini sampel teks yang digunakan berasal dari Wikipedia.

Silahkan Anda cari saja definisi “Regular Expression” di Wikipedia berbahasa Bahasa Inggris.

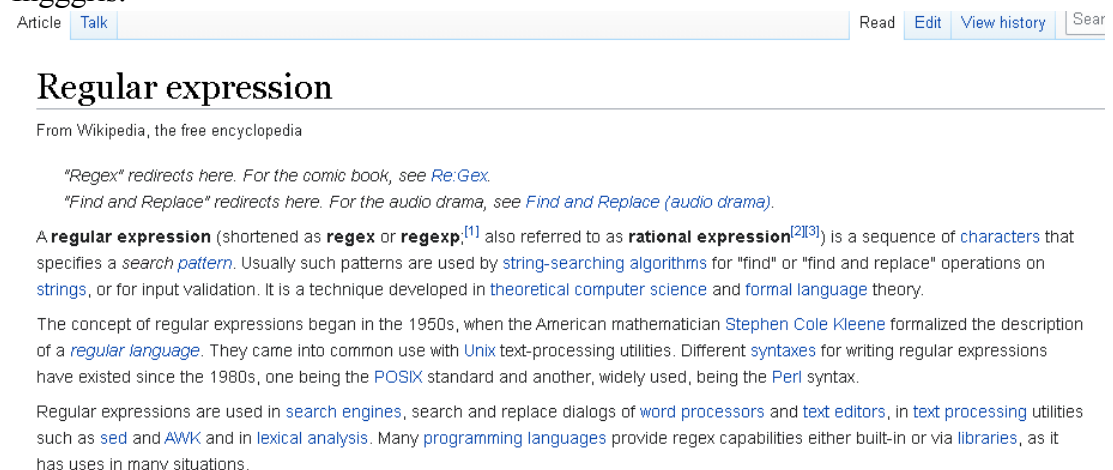


Figure 2. Hasil Pencarian pada Wikipedia

Figure 2 menunjukkan hasil pencarian tentan Regular Expression pada Wikipedia berbahasa Inggris.

3. Salin Sampel Teks ke Text String

Salinlah sampel teks yang Anda temukan tersebut ke kotak isian “Text String” pada Regexpal.

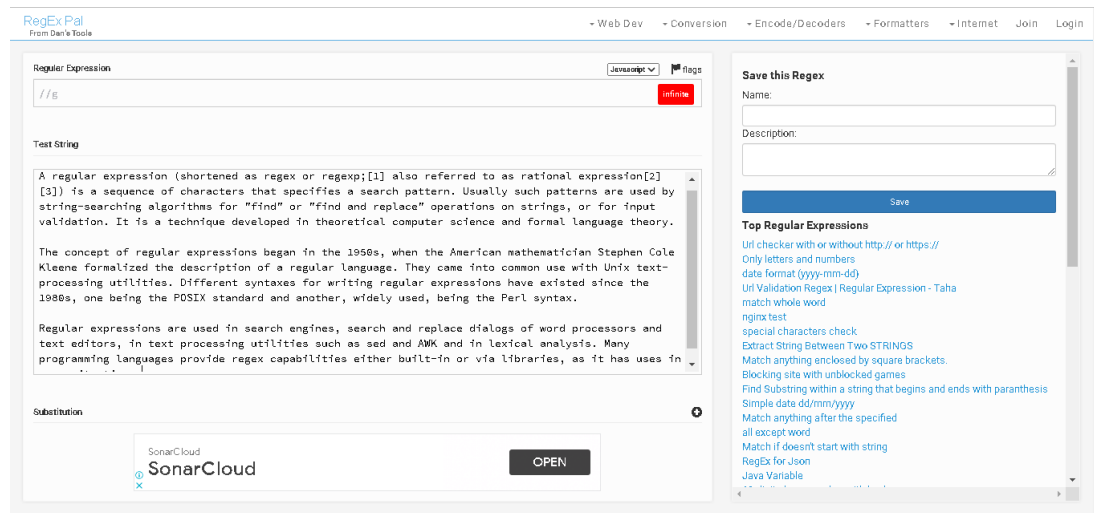


Figure 3 Teks yang Disalin

Figure 3 menunjukkan sampel teks yang sudah disalin pada Regexpal. Anda masih dapat mengeditnya sesuai dengan kebutuhan Anda.

4. Masukkan Query RegEx

Setelah semua siap, silahkan Anda masukkan query Regular Expression sesuai dengan keinginan Anda. Namun, agar praktikum ini terarah silahkan coba beberapa query berikut dan lihat apa hasilnya. Perlu diperhatikan, Regexpal akan menandai hasil pencarian dengan blok warna biru pada Text String.

- use
- [use]
- []
- \\[]
- The
- ^The
- situations\\. \$
- and*
- and+
- s{2}
- \\d
- \\D
- \\s
- \\S
- \\w
- \\W
- [a-l][a]
- [0-9][0-9][0-9][0-9]
- [a-zA-Z]

C. Tugas Praktikum

Pada tugas praktikum kali ini Anda ditugaskan untuk mencoba menerapkan teori Regular Expression ke dalam Bahasa Python. Anda dibebaskan untuk menentukan contoh ataupun studi kasusnya. Berikut ketentuan-ketentuan yang harus Anda perhatikan.

1. Tugas dikerjakan pada Jupyter Notebook.
2. Nama file Jupyter Notebook Anda adalah `Regex_NIM.ipynb`
3. Berikan identitas berupa Nama, NIM, Kelas pada header Jupyter Notebook Anda
4. Berikan deskripsi yang jelas pada masing-masing tahapan atau contoh yang Anda lakukan.
5. Tugas dikumpulkan pada kuliahonline.amikompurwokerto.ac.id pada link yang disediakan pada folder Minggu #2.
6. Deadline pengumpulan adalah maksimal 6 hari setelah tugas ini diberikan.

Yang dinilai dari tugas ini adalah:

1. Kesesuaian format dengan ketentuan yang diberikan (10%)
2. Ketepatan waktu pengerjaan (10%)
3. Kelengkapan dan kejelasan pembahasan (80%)