RICE UNIVERSITY

Similar to (regEx)

This presentation may be shared with current and future students enrolled in COMP 430 / 533.

Dong-Lin Wu, Rung-De Chu, 11/21/2023

Slide & Jupyter Notebook :

https://is.gd/1l3ww4

What is Regular Expression (regex) ?

- Sequences of characters that form a search pattern
- Used in programming and text processing for tasks like **searching, replacing**, and **validating data**.
- Flexibility, Efficiency & Powerful
- Use Cases :
  - Validating email addresses.
  - Searching for specific patterns in log files.
  - Extracting information from text data.

Syntax of Regular Expression

- **Literals:** Ordinary characters as exact matches
  - ‘a’, ‘1’, ‘!’
- **Metacharacters:** Special characters
  - ‘\m’ and ’\M’ : Start/ end of a word
  - ‘ * ’ : Zero or more occurrences ‘lo*’ : can be ‘l’, lo’ , ‘loooo’
  - ‘+’ : One or more occurrences. ’lo+ ‘ : can be ‘lo’ ,’loooo’
  - ‘ ? ‘ : Zero or one occurrences. ‘lo?” can be ‘l’ or ‘lo’

Syntax of Regular Expression

- **Character Classes** **[]** : matches any character inside
  - [abc] matches ‘a’ , ‘b’ or ‘c’
- **Range** : Use - to specify a range within character classes.
  - [a-z] , [0-9]
- **Escape Character**: \ : turns metacharacters into literals.
  - \. matches a literal dot.
- **Alternation** | : for logical OR
  - cat|dog

Syntax of Regular Expression

- **Quantifiers:** Dictate the frequency of the preceding element.
  - {n}: Exactly n times. a{3} : ‘aaa”
  - {n,}: n or more times. a{2,}: ‘aa’,’aaa’,aaaa’,
  - {n,m}: Between n and m times. a{2,4} : 'aa', 'aaa', or 'aaaa'.
- **Groups and Capturing**: () : groups parts of the pattern.
  - (abc){2} will match 'abcabc'.
- **Wildcard**: _ for any single character, % for any string

Similar To

- compare a string against a pattern
- **string [NOT] SIMILAR TO pattern [ESCAPE escape-character]**

  Return true if the string match/not matches the pattern
- Escape Character:
  - A backslash \ is used to negate the special meaning of metacharacters.
  - A different escape character can be specified using the ESCAPE clause.
  - Disable the escape capability by writing ESCAPE ‘ '.

Users Table

Given a table containing various individual fields such as SSN, lastname, middlename, firstname, and phone.

- Input: (SSN, lastname, middlename, firstname, phone)

  ('123-45-6789', 'Smith', '', 'Adam', '713-555-0101’),

  ('345-67-8901', 'Williams', '', 'James', '408-555-0102’),

  ('567-89-0123', 'Jones', '', 'Michael', '332-555-0103’),.....

Example 1

Given a table mentioned above, return the records of individuals whose last name is exactly five characters long with 'o' as the third letter.

- Sample Output:

  ('456-78-9012', 'Brown', 'Chris', 'Emily', '408-555-0117'),

  ('162-34-5684', 'Moore', 'Sue', 'Isabella', '973-555-0124')

- regEx

  _ _ o _ _

Hint:

Wildcard: _ for any single character, % for any string

Example 1 Solution

```
SELECT *
FROM Users
WHERE
     lastname SIMILAR TO '__o__';
```

Expected Output:

ssn	lastname	middlename	firstname	phone
456-78-9012	Brown	Chris	Emily	408-555-0117
162-34-5684	Moore	Sue	Isabella	973-555-0124

Example 2

Given a table mentioned above, return the records of individuals whose last name is ‘Taylor” or last name starting with J or W

- Sample Output:

  (‘345-67-8901’, ‘Williams', ‘ ', ‘James', '408-555-0102’),

  (‘234-56-7890', ‘Johnson', ‘Lee', ‘Maria’,’ 713-555-0116’)

- regEx

  Taylor | \mJ | \mW

Hint:

‘\m’ and ’\M’ : Start/ end of a word

Example 2 Solution

Solution 1 : SIMILAR TO syntax

```
SELECT *
FROM Users
WHERE
    lastname SIMILAR TO 'Taylor|\mJ%|\mW%';
```

Solution 2 : LIKE syntax

```
SELECT *
FROM Users
WHERE
    lastname = 'Taylor' OR
    lastname LIKE 'J%' OR
    lastname LIKE 'W%';
```

Expected Output :

ssn	lastname	middlename	firstname	phone
345-67-8901	Williams		James	408-555-0102
567-89-0123	Jones		Michael	332-555-0103
132-34-5681	Taylor		Justin	408-555-0107
234-56-7890	Johnson	Lee	Maria	713-555-0116
102-34-5678	Wilson	Ray	Amanda	564-555-0120
122-34-5680	Taylor	Jane	Patricia	713-555-0121
202-34-5688	White	Gail	Harper	332-555-0123
242-34-5692	Taylor	Zoe	Elizabeth	973-555-0129

Try It !

Since the data is messy and potentially contains errors in various fields, return the validate data with specific format for this table.

- Data Requirement :
  - **SSN:** xxx-xx-xxxx digit only
  - **phone:** xxx-xxx-xxxx digit only
  - **firstname and lastname:** alphabetic character, optionally including hyphens, apostrophes, or spaces.
  - **middlename:** same as firstname and lastname, but can be empty

Slide & Jupyter Notebook :

https://is.gd/1l3ww4

Try It !

- Input: (SSN, lastname, middlename, firstname, phone)

  ('412523324', 'Smith', '', 'Adam', '713-555-0101'), -- wrong ssn

  ('524-6575-43', 'Williams', '', 'James', '408-555-0102'), -- wrong ssn

  ('657-43-2345', '@Jones', '', 'Michael', '332-555-0103'), -- wrong last name

  ('427-52-3455', 'Davis', '123', 'Karen', '973-555-0104'), -- wrong middle name

  ('154-35-3453', 'Johnson', '', 'Emily', '2125550198'), -- wrong phone

  ('154-35-3423', 'Liu', ' ' , 'Guan-Yu', '564-555-0105'),

  ('987-65-4321', 'O''Brien', 'Patrick', 'James', '415-555-0234'),

  ('456-78-9123', 'Davis', 'Anne', 'Michael', '305-555-0177');

- Output:

  ('154-35-3423', 'Liu', ' ' , 'Guan-Yu', '564-555-0105'),

  ('987-65-4321', 'O''Brien', 'Patrick', 'James', '415-555-0234'),

  ('456-78-9123', 'Davis', 'Anne', 'Michael', '305-555-0177');

Try It Solution

```
SELECT * FROM SourceUsers
WHERE
    ssn SIMILAR TO '[0-9][0-9][0-9]-[0-9][0-9]-[0-9][0-9][0-9][0-9]' AND
    lastname SIMILAR TO '[A-Za-z]+([-'' ]?[A-Za-z]+)*' AND
    firstname SIMILAR TO '[A-Za-z]+([-'' ]?[A-Za-z]+)*' AND
    middlename SIMILAR TO '([A-Za-z]+([-'' ]?[A-Za-z]+)*)*' AND
    phone SIMILAR TO '[0-9][0-9][0-9]-[0-9][0-9][0-9]-[0-9][0-9][0-9][0-9]';
```

Expected Output:

ssn	lastname	middlename	firstname	phone
154-35-3423	Liu		Guan-Yu	564-555-0105
987-65-4321	O'Brien	Patrick	James	415-555-0234
456-78-9123	Davis	Anne	Michael	305-555-0177

Any Questions?

Contact info : dw73@rice.edu, rc118@rice.edu