JavaScript ES6

Lesson 8.Working With Regular Expressions

Lesson Objectives

After completing this lesson, you will be able to:

- Use regular expressions
- Search text using simple patterns and special characters

Work with RegExp objects

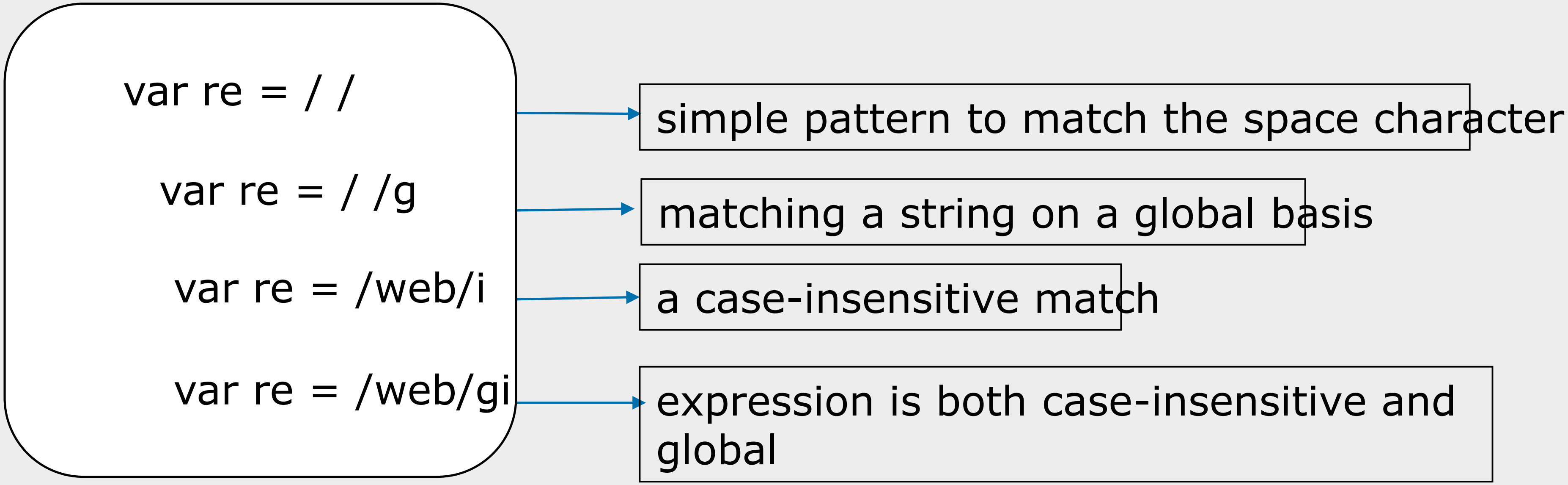
Regular Expressions

Sequence or pattern of characters, matched against a text string, when you perform searches and replacements

Perform client-side data validations or any other extensive text entry parsing

RegEx – Simple Patterns

A simple regular expression uses no special characters for defining the string to be used in a search



```
var re = / /
var re = / /g
var re = /web/i
var re = /web/gi
```

simple pattern to match the space character

matching a string on a global basis

a case-insensitive match

expression is both case-insensitive and global

RegEx – Special Characters

\b Word Boundary:

- Get a match at the beginning or end of a word in the string
- /\bor/ matches “origami” and “or” but not “normal”.
- /or\b/ matches “traitor” and “or” but not “perform”
- /\bor\b/ matches full word “or” and nothing else

\B Word Non-Boundary:

- Get a match when it is not at the beginning or end of a word in the string
- /\Bor/ matches “normal” but not “origami”
- /or\B/ matches “normal” and “origami” but not “traitor”
- /\Bor\B/ matches “normal” but not “origami” or “traitor

RegEx – Special Characters (Contd.)

\d Numeral:

- Find any single digit 0 through 9
  - /\d\d\d/ matches “212” and “415” but not “B17”

\D Non-numeral:

- Find any non-digit
  - /\D\D\D/ matches “ABC” but not “212” or “B17”

\s Single White Space:

- Find any single space character
  - /over\sbite/ matches “over bite” but not “overbite” or “over bite”

RegEx – Special Characters (Contd.)

\S Single Non-White Space:

- /over\Sbite/ matches “over-bite” but not “overbite” or “over bite”

\w Letter, Numeral, or Underscore:

- /A\w/ matches “A1” and “AA” but not “A+”

\W Not letter, Numeral, or Underscore:

- /A\W/ matches “A+” but not “A1” and “AA”

RegEx – Special Characters (Contd.)

"." Any Character Except Newline:

- /…/ matches "ABC", "1+3", "A 3" or any 3 characters

[…] Character Set:

- Finds any character in the specified character set
  - /[AN]BC/ matches "ABC" and "NBC"

[^…] Negated Character Set:

- Find any character not in the specified character set
  - /[^AN]BC/ matches "BBC" and "CBC" but not "ABC" or "NBC"

RegEx – Counting Metacharacters

“*” - Zero or More Times:

- /Ja*vaScript/ matches “JvaScript”, “JavaScript”, and “JaaavaScript” but not “JovaScript”

“?” - Zero or One Time:

- /Ja?vaScript/ matches “JvaScript” or “JavaScript” but not “JaaavaScript”

“+” - One or More Times:

- /Ja+vaScript/ matches “JavaScript” or “JaavaScript” but not “JvaScript”

RegEx – Counting Metacharacters (Contd.)

{n} - Exactly n Times:

- /Ja{2}vaScript/ matches “JaavaScript” but not “JvaScript” or “JavaScript”

{n,} - N or More Times:

- /Ja{2,}vaScript/ matches “JaavaScript” or “JaaavaScript” but not “JavaScript”

{n,m} - At Least n, At Most m Times:

- /Ja{2,3}vaScript/ matches “JaavaScript” or “JaaavaScript” but not “JavaScript”

RegEx – Positional Metacharacters

“^” - At the beginning of a string or line

- /^Fred/ matches “Fred is OK” but not “I’m with Fred” or “Is Fred here?”

“$” - At the end of a string or line

- /Fred$/ matches “I’m with Fred” but not “Fred is OK” or “Is Fred here?”

Regular Expression Object

```
regExpObject = /pattern/ [g | i | gi]
    regExpObject = new RegExp([“pattern”, [“g”|“i”|“gi”]])
```

global	ignoreCase
lastIndex	source

Regular Expression Object (Contd.)

compile(“pattern”, [“g” | “i” | “gi”])
test(“string”)
exec(“string”)

```
var re = / somePattern/

var matchArray = re.exec(“someString”)
```

Demo

Test_compiler.html
DemoRegExp.html

Lab

Lab Exercise 9:

- Regular Expressions in JavaScript

Summary

For client-side data validation we can use a regular expression

Regular expression object describes a pattern of characters

Simple regular expressions use no special characters used to match the space in a string with an underscore character

Regular Expressions use special characters such as \b, \d, \w etc

Review Questions

Question 1: The ___________ property is the main string against which a regular expression is compared in search of a match.

- Option 1: RegExp.input
- Option 2: RegExp.inp
- Option 3: RegExpr.input

Question 2: Index property indicates the index counter of the main string to be searched against the current regular expression object.

- True / False

Question 3: Use the ____________ method to compile on the fly a regular expression whose content changes continually during the execution of a script.

Match the Following

1 . \b
2. \B
3 . \d
4 . \s
5 . \S

a. Word non-boundary
b. Word boundary
c. Numeral
d. Single non-white space
e. Single white space