# JavaScript Homework #3 - RegEx

Please study following RegExp patterns and find out why:

// example 1

const myString = "99 bottles of beer on the wall " +

"take 1 down and pass it around -- " +

"98 bottles of beer on the wall.";

const match = myString.match(/[\-0-9a-z.]/ig);

const match = myString.match(/[^\-0-9a-z.]/);

const match = myString.match(/[0-9][0-9][0-9]|[0-9][0-9]|[0-9]/);

const match = myString.match(/[0-9]+/);

// example 2

const stuff =

'hight: 9\n' +

'medium: 5\n' +

'low: 2\n';

const levels = stuff.match(/:\s\*[0-9]/g);

// example 3

const messyPhone = '(505) 555-1515';

const neatPhone = messyPhone.replace(/\D/g, '');

// example 4

const field = ' something ';

const valid = /\S/.test(field);

// example 5

const input = "Address: 333 Main St., Anywhere, NY, 55532. Phone: 555-555-2525.";

const match = input.match(/\d{5}.\*/);

const equation = "(2 + 3.5) \* 7";

const match = equation.match(/\(\d \+ \d\.\d\) \\* \d/);

// example 6

const text = "Visit oreilly.com today!";

const match = text.match(/[a-z]+(?:\.com|\.org|\.edu)/i);

// example 7

const promo = "Opening for XAAX is the dynamic GOOG! At the box office now!";

const bands = promo.match(/(?:[A-Z])(?:[A-Z])\2\1/g);

// example 8

const input = "One two three";

input.replace(/two/, '($`)'); // "One (One ) three"

input.replace(/\w+/g, '($&)'); // "(One) (two) (three)"

// example 9

const input = "It was the best of times, it was the worst of times";

const beginning = input.match(/^\w+/g); // "It"

const end = input.match(/\w+$/g); // "times"

const everything = input.match(/^.\*$/g); // sames as input

const nomatch1 = input.match(/^best/ig);

const nomatch2 = input.match(/worst$/ig);

// example 10

const input = "One line\nTwo lines\nThree lines\nFour";

const beginnings = input.match(/^\w+/mg); // ["One", "Two", "Three", "Four"]

const endings = input.match(/\w+$/mg); // ["line", "lines", "lines", "Four"]

// example 11

const inputs = [

"john@doe.com", // nothing but the email

"john@doe.com is my email", // email at the beginning

"my email is [john@doe.com](mailto:john@doe.com)", // email at the end

"use john@doe.com, my email", // email in the middle, with comma afterward

"my email:john@doe.com.", // email surrounded with punctuation

];

const emailMatcher =

/\b[a-z][a-z0-9.\_-]\*@[a-z][a-z0-9\_-]+\.[a-z]+(?:\.[a-z]+)?\b/ig;

inputs.map(s => s.replace(emailMatcher, '<a href="mailto:$&">$&</a>'));

// returns [

// "<a href="mailto:john@doe.com">john@doe.com</a>",

// "<a href="mailto:john@doe.com">john@doe.com</a> is my email",

// "my email is <a href="mailto:john@doe.com">john@doe.com</a>",

// "use <a href="mailto:john@doe.com">john@doe.com</a>, my email",

// "my email:<a href="mailto:john@doe.com>john@doe.com</a>.",

// ]