

1.create a form, do the validation with the help of jquery and regular expression.

i>Username should only start with letter A,P,Q,Z and end with Biya,We,Jzz

```
<!-- /.container -->
<script>
    function validateForm() {
        var fname = $('#first_name').val();
        var fnameRx = /^[APQZ]\w*(Biya|We|Jzz)$/;
        if (!fnameRx.test(fname)) {
            $('#error').show();
            $('#error').text("Invalid name, name should start with A, P, Q or Z and end with Biy
            //return true;
        }
    }
}
```

ii>User the last name should start with Bhfy, W124, Jv.! and end with A, P,Q,Z

```
//For lastname
var lname = $('#last_name').val();
var lnameRx = /^(Bhfy|W124|Jv.!)\w*([APQZ])$/;
if (!lnameRx.test(lname)) {
    $('#error').show();
    $('#error').text("Invalid name, last name should start with Bhfy, W124, Jv.! and end with A,
    //return false;
}
```

iii> Contact no should start with 9,6,7,8 and end with 7,3646,976

```
//For Contact Number

var contact_no = $('#contact_no').val();
var contact_noRx = /^[9678]\d{1,7}(7|3646|976)$/;
// var contact_noRx = /^[a-zA-Z]+[@][A-Za-z][.][a-zA-Z])$/;
if (!contact_noRx.test(contact_no)) {
    $('#error').show();
    $('#error').text("Contact no should start with 9,6,7,8 and end with 7,3646,976");

    return false;
}
```

iv> Email validation should be there with regular expression.

```
//For email
var email = $('#email').val();
var emailRx = /^[a-zA-Z0-9_\-\.\.]+@([a-zA-Z0-9_\-\.\.]+\.)?([a-zA-Z]{2,5})$/;
if (!emailRx.test(email)) {
    $('#error').show();
    $('#error').text("Invalid Email");

    return false;
}
```

*\*The above exercise will be on HTML and jquery with the proper functioning*

2. Use XPath to determine/Select At least 10 elements on google chrome from the given HTML:-

[https://docs.google.com/document/d/14O5UaTngsxnSIbYYVwR3y8THNnGBJXx0y\\_DtQdivAkU/edit](https://docs.google.com/document/d/14O5UaTngsxnSIbYYVwR3y8THNnGBJXx0y_DtQdivAkU/edit)

*\*do not copy XPath from developer tool use only your understanding to determine Xpath*

```
console.log($(_x('//h1[@class="header"]/a[2]')).css({
  'color': 'green',
  'border': '3px solid red'
}));
console.log($(_x('//*[@class="main"]/h1/a')).css({
  'color': 'green',
  'border': '3px solid red'
}));
console.log($(_x('/html/body/div/div/p[1]')).css({
  'color': 'green',
  'border': '3px solid red'
}));
console.log($(_x('//table[@id="tableOfFruits"]//tr[@class="title"]/td')).css({
  'color': 'green',
  'border': '3px solid red'
}));
$_x('//*[@tr[@class="heading"]/td')).css({
  'color': 'green',
  'border': '3px solid red'
});
```

```

console.log($('x(/html/body/div/table/tbody/tr[2]/td[1]')).css({
  'color': 'green',
  'background-color': 'red'
}));
console.log($('x(/html/body/div/table/tbody/tr[4]/td[1]')).css({
  'color': 'green',
  'background-color': 'red'
}));
console.log($('x(/html/body/div/table/tbody/tr[6]/td[1]')).css({
  'color': 'green',
  'background-color': 'red'
}));
console.log($('x(/html/body/div/table/tbody/tr[8]/td[1]/table/tbody/tr/td')).css({
  'color': 'green',
  'border': '3px solid red'
}));
console.log($('x(/html/body/div/table/tbody/tr[10]/td[1]')).css({
  'color': 'green',
  'border': '3px solid red'
}));

```

Activities Google Chrome Mon 10:19

Complex Table Sort | Java x New Tab

Not secure | example1.com/regex/xpath2.html

Apps Gmail YouTube Maps History

## UIZE JavaScript Framework

### JAVASCRIPT EXAMPLES

#### Complex Table Sort

#### SOURCE

In this example, an instance of the [Uize.Widget.TableSort](#) widget class is being spawned and attached to a table of data. This use of the `Uize.Widget.TableSort` class is notable in that the table's markup is rather complex in a number of ways. Firstly, there is a separator table row between each row of data. Secondly, there is a separator row between every three rows of data. When you click the column headings to sort different columns, notice how the separator rows are not included in the sort and do not budge. To add another wrinkle, the fruit names in the first column are buried deep inside nested tables, yet the text is still found and sorting by fruit works as expected. Finally, there is a `thead` tag that contains the title for the table, and the table's data is contained inside the `tbody` tag. None of these little kinks throw off the table sorting logic.

Fruit Nutritional Information (per 8oz serving)										
Fruit	Calories	Total Fat	Sat. Fat	Salt	Total Carbs	Fiber	Sugars	Protein	Calcium	Potassium
Apples	52 kcal	0.17 g	0.028 g	1 mg	13.81 g	2.4 g	10.39 g	0.26 g	6 mg	107 mg
Avocados	160 kcal	14.66 g	2.126 g	7 mg	8.53 g	6.7 g	0.66 g	2 g	12 mg	485 mg
Bananas	89 kcal	0.33 g	0.112 g	1 mg	22.84 g	2.6 g	12.23 g	1.09 g	5 mg	358 mg
Dates	277 kcal	0.15 g	0 g	1 mg	74.97 g	6.7 g	66.47 g	1.81 g	64 mg	696 mg
Grapefruits	42 kcal	0.14 g	0.021 g	0 mg	10.66 g	1.6 g	6.89 g	0.77 g	22 mg	135 mg
Grapes	67 kcal	0.35 g	0.114 g	2 mg	17.15 g	0.9 g	16.25 g	0.63 g	14 mg	191 mg
Mangos	65 kcal	0.27 g	0.066 g	2 mg	17 g	1.8 g	14.8 g	0.51 g	10 mg	156 mg

3. In the given HTML use **text**, **contains** the function of XPATH to determine the value of different 5 fruit names.

```
// //console.log($_x('//td[contains(text(), "Orange")]').text());
var obj = $_x('//*[contains(text(), \"Apples\")]')).css({
  'color': 'red'
});
console.log(obj[0]);
console.log(obj[0].innerText);
var obj1 = $_x('//*[contains(text(), \"Avocados\")]')).css({
  'color': 'green'
});
console.log(obj1[0]);
console.log(obj1[0].innerText);
var obj2 = $_x('//*[contains(text(), \"Bananas\")]')).css({
  'color': 'yellow'
});
console.log(obj2[0]);
console.log(obj2[0].innerText);
var obj3 = $_x('//*[contains(text(), \"Dates\")]')).css({
  'color': 'brown'
});
console.log(obj3[0]);
console.log(obj3[0].innerText);
var obj4 = $_x('//*[contains(text(), \"Grapefruits\")]')).css({
  'color': 'teal'
});
console.log(obj4[0]);
console.log(obj4[0].innerText);
```

Apples	52 kcal	0.17 g	0.028 g	1 mg	13.81 g	2.4 g	10.39 g	0.26 g	6 mg	107 mg
Avocados	160 kcal	14.66 g	2.126 g	7 mg	8.53 g	6.7 g	0.66 g	2 g	12 mg	485 mg
Bananas	89 kcal	0.33 g	0.112 g	1 mg	22.84 g	2.6 g	12.23 g	1.09 g	5 mg	358 mg
Dates	277 kcal	0.15 g	0 g	1 mg	74.97 g	6.7 g	66.47 g	1.81 g	64 mg	696 mg
Grapefruits	42 kcal	0.14 g	0.021 g	0 mg	10.66 g	1.6 g	6.89 g	0.77 g	22 mg	135 mg



4.create database according to your understanding with basic profile detail and write queries to fetch all data-name, contact, email, address on the basis of REGULAR EXPRESSION.

TABLE

```
mysql> select * from form;
```

first_name	last_name	email	contact_no
AaaaBiya	BhfyZ	ef@ef.com	9209348309
AaaaFbiya	BhfyGHGZ	eram45@gmail.com	7657686464
AaaaBiya	BhfyZ	eram@gmail.com	7657646464
eram	Fatima	erum@gmail.com	9891165336
AaaddtdtgaBiya	BhfyhdtgdZ	erumhdttdtd@yopmail.com	1212121212
AaaaBiya	BhfyhdddddfdfZ	sahilqaiser7856@hmm.com	1234512345
sadaf	fatima	sahilqaiser@hmm.com	8209348309
Aaaammmbiya	BhfymmmZ	sahilqaiser@hmmnnn.com	9767676976
sara	khan	sara@example.com	8765263748

9 rows in set (0.00 sec)

## QUERIES USING REGEXP

```
mysql> SELECT * FROM form WHERE first_name REGEXP '^(Aaa)+';
```

first_name	last_name	email	contact_no
AaaaBiya	BhfyZ	ef@ef.com	9209348309
AaaaFbiya	BhfyGHGZ	eram45@gmail.com	7657686464
AaaaBiya	BhfyZ	eram@gmail.com	7657646464
AaaddtdtgaBiya	BhfyhdtgdZ	erumhdttdtd@yopmail.com	1212121212
AaaaBiya	BhfyhdddddfdfZ	sahilqaiser7856@hmm.com	1234512345
AaaammBiya	BhfymmmZ	sahilqaiser@hmmnnn.com	9767676976

```
6 rows in set (0.00 sec)
```

```
mysql> select * from form where email REGEXP '^(er)+';
```

first_name	last_name	email	contact_no
AaaaFbiya	BhfyGHGZ	eram45@gmail.com	7657686464
AaaaBiya	BhfyZ	eram@gmail.com	7657646464
eram	Fatima	erum@gmail.com	9891165336
AaaddtdtgaBiya	BhfyhdtgdZ	erumhdttdtd@yopmail.com	1212121212

```
4 rows in set (0.00 sec)
```

```
mysql> mysql> SELECT * FROM form WHERE last_name REGEXP '^(BhfyZ)+';
```

first_name	last_name	email	contact_no
AaaaBiya	BhfyZ	ef@ef.com	9209348309
AaaaBiya	BhfyZ	eram@gmail.com	7657646464

```
2 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM form WHERE contact_no REGEXP '^(9209)+';
```

first_name	last_name	email	contact_no
AaaaBiya	BhfyZ	ef@ef.com	9209348309

```
1 row in set (0.00 sec)
```

