



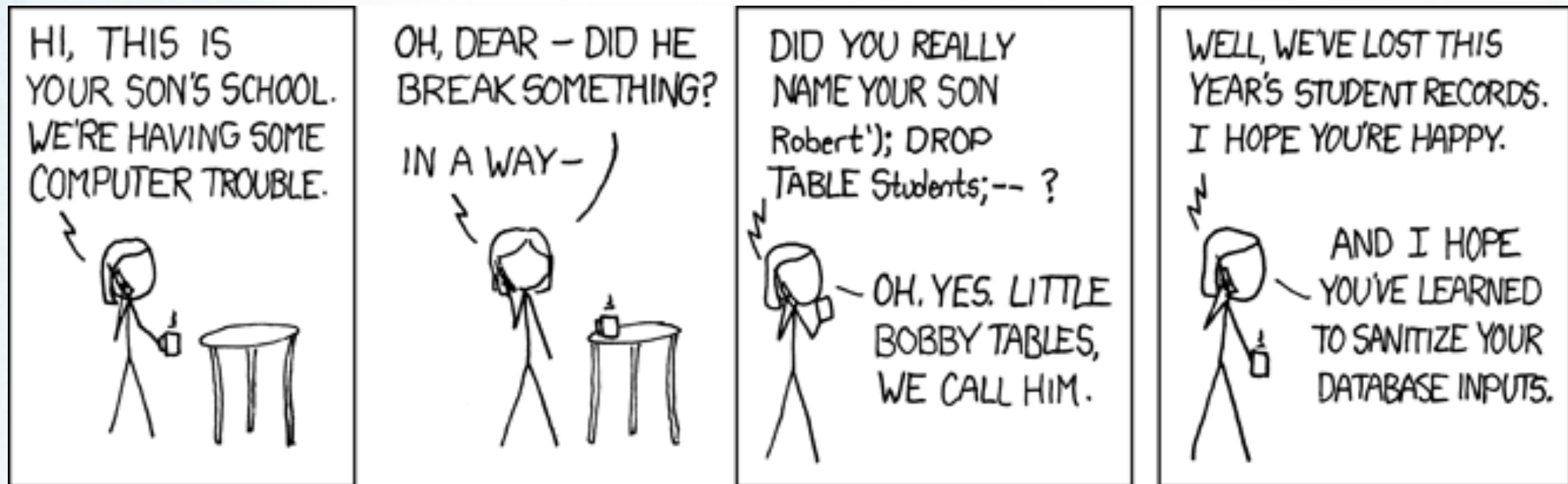
НИ Е ВЯРВАМЕ ВЪВ ВАШЕТО БЪДЕЩЕ

JavaScript form validation and RegEx

Form Validation

Защо е необходимо

- На първо място: "Защото така се прави!"
- Подобряваме юзер експириънса - представете си че някой въведе грешен имейл при регистрация и ние го приемем без да го валидираме
- .. или пък че "парола" и "потвърди парола" не съвпадат
- Погледнато реалистично - валидация на инпут-а трябва да се прави абсолютно винаги защото:
 - после ние (програмистите) трябва да използваме тези данни в нашия уебсайт/приложение
 - това е един от най-разпространените начини за хак-ване на дадена система (чрез инпут параметри)



<http://www.smashingmagazine.com/2011/01/keeping-web-users-safe-by-sanitizing-input-data/>

Какво е валидация

- Валидация, това е проверката и потвърждението, че нещото което валидираме (някакъв инпут) е точно това, което очакваме то да бъде
- Например ако плащате в магазина, касиера винаги проверява на първо време дали му давате пари, после дали сте му дали достатъчно и накрая - дали не сте му дали в повече.
- Това е валидация.
- При това ако първите две проверки не "минат" , бъдете сигурни че ще се предизвика грешка, която ще бъде обработена от касиера или в краен случай от охраната на магазина.
- В третия случай - вашият инпут (сумата, която давате) ще "премине" успешно валидацията и касиера просто ще ви върне ресто.

Какво валидираме най-често

- Пароли
- имейл
- текст (например дали съдържа "специални" символи)
- телефонни номера
- номера на сметки или кредитни карти
- възраст, дати

Как достъпваме form елементите през JS

- Използваме `$(form).submit()` event-a
- Можем да получим form елемента през самия event:

```
$('#myForm').submit(function (e) {  
    var form = e.target;  
});
```

- или дори още по-лесно, чрез служебната променлива **this**:

```
$('#myForm').submit(function (e) {  
    var form = this;  
});
```


Как достъпваме form елементите през JS

- Веднъж след като имаме форм елемента, можем да достъпим всички инпут полета от формата по тяхните имена:

```
var form = e.target;  
var emailInput = $(form).find('input[name=email]');  
var email = emailInput.val();
```

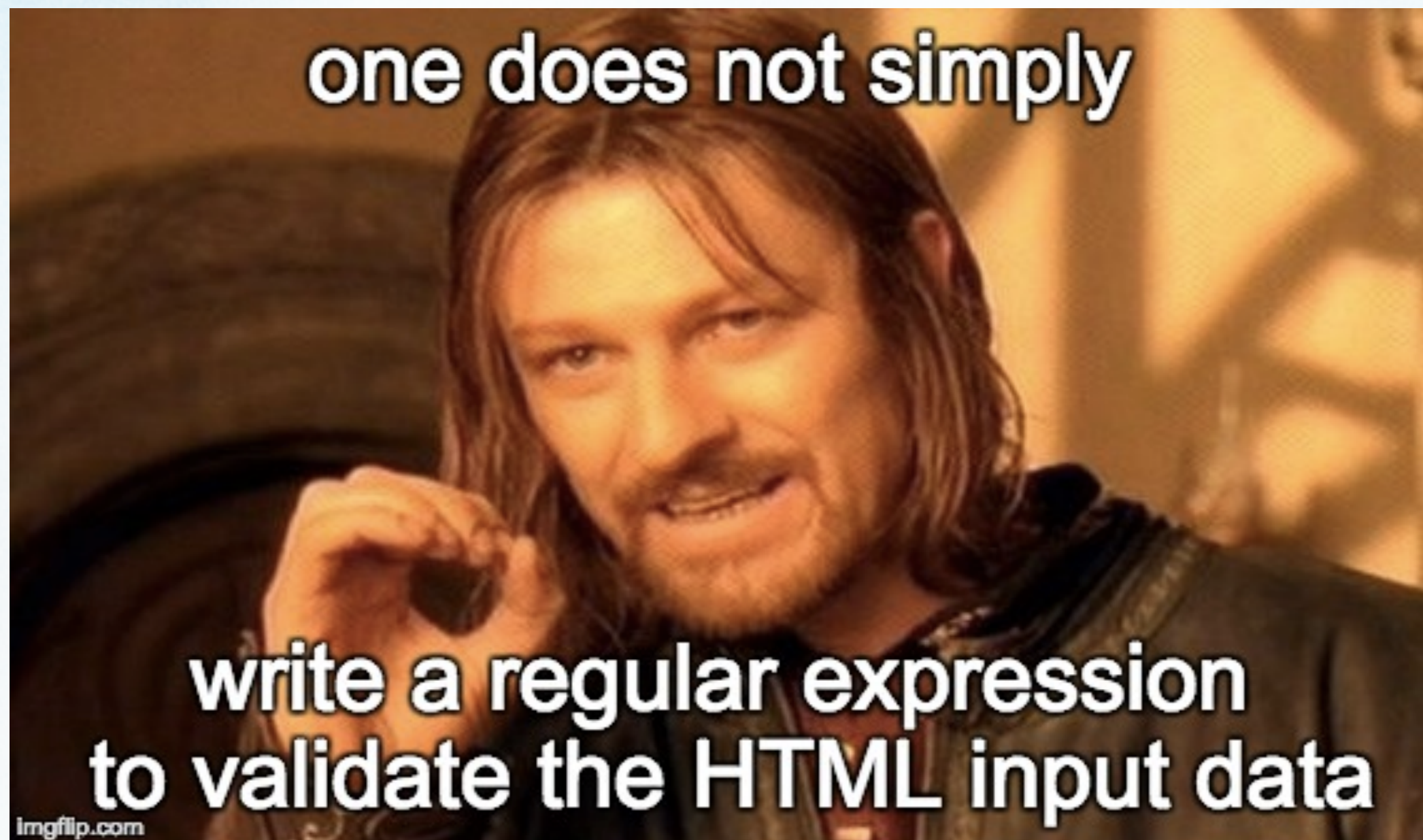
- Не трябва също да забравяме и да предотвратим дефолт-ното събмитване на формата:

```
$('#myForm').submit(function (e) {  
    e.preventDefault();  
});
```

Въпроси?

Примери

RegEx



Какво представляват регулярните изрази

- Това са шаблони за текст
- В тях използваме неща като "*" или други специални знаци, на местата където искаме да ни съвпадат повече от една букви от текста
- Например:

шаблona

`/how are you?+ /`

ще съвпадне с:

`"how are you?"` и с `"how are you???"`, и с `"how are you???????"`

И още примери

- шаблона: `/I love .*!/?`
- ще съвпадне с:
"I love pizza!" и с "I love ponies!"
- както и със:
"I love pizza and ice cream, and football, and beer!"
- но няма да съвпадне със:
"I don't love baseball!", "I like blues!" и "I LOVE you!"

- Т.е. те ни помагат да проверим дали даден текст "пасва" на даден шаблон (pattern matching)
- шаблона може да е всичко - от това дали стринга да се състои само от букви или само от цифри, до това колко да е дълъг, какви символи да има в него и на какви позиции
- Съответно, за да проверим дали един email е валиден, трябва да си направим шаблон от типа на:

<текст>@<текст>.<текст>

- нека пробваме ето така: `email.match(/w+@w+\.w+/)`

Как да разгадаем тези кодове и знаци?

- https://developer.mozilla.org/en/docs/Web/JavaScript/Guide/Regular_Expressions
- <https://www.debuggex.com/cheatsheet/regex/javascript>
- Основното, което трябва да знаем е:
 - регулярният израз се изписва между две наклонени черти:
`/the-regex-comes-here/`
 - Всеки интервал или препинателен знак има значение!!
`/match/ != / match /`
 - Символите: `!, ?, ., -, [,], (,), {, }, |, *, +, ^, $, \` и `/` са специални и се използват за маркиране на различни видове текст в стринг-а

Примери

```
mobilePhone.match (/^(+|00)359 ?\d{3} ?\d{3} ?\d{3}$/)
```

```
birthDate.match (/^\d\d\d\d-\d\d-\d\d$/)
```

```
birthDate.match (/^(19\d\d|20(0\d|1[0-5]))-[01]?\d-[0123]?\d$/)
```

```
cardNumber.match (/^\d{13}$/)
```

```
name.match (/^[A-Z][a-z]+ [A-Z][a-z]+$/)
```

```
url.match (/^https?:\/\/\w+(\w+\.)+\/)
```

```
email.match (/^\w[\w.-]+@(\w+\.)+\w+$/)
```


Въпроси?

Примери

<http://zenifytheweb.com/courses/lessons/lesson15/example/index.html>

Домашно

<https://github.com/zzeni/swift-academy-homeworks/tree/master/tasks/L15>