

Front End Frameworks

Why

As your code becomes exponentially huge, you can end up with massive files and have no idea where code is located. You end up making the code exponentially more complicated and have no idea what code affects other code

Enter.... Javascript Frameworks

Don't reinvent the wheel

- understand how its working
- don't spend a long time figuring it out
- DOM selection is a great example

Size concerns

what happens when you have large dev teams? should everyone touch the same code?

Code separation

- Using a framework allows/forces us to implement a separation of concerns
- Each bit of code that relates to different things is on its own
- this is also called a Design Pattern

```
$.get("http://calendar-server.elasticbeanstalk.com/messages", function(data, status) {  
    $('ul').empty();  
    data = _.difference(data, messages);  
    messages = messages.concat(data);  
    var lastTen = messages.slice(messages.length - 5, messages.length);  
    lastTen.forEach(function(msgObject) {  
        $('<li>').attr('class', 'collection-item grey lighten-4')  
            .html('<h5 class="light-blue-text">' + msgObject.created_by + '</h5>' + msgObject.message)  
            .appendTo('ul');  
    });  
});
```

Horrible Seperation of Concerns

MVC

One of the most common design patterns is MVC

- Model
- View
- Controller

Frameworks

- React
- Angular
- Ember
- Backbone