

# Authoring jQuery Plugins

Joe Eames

@josepheames

[www.pluralsight.com](http://www.pluralsight.com)



# Outline

- **Naming**
- **Basic Plugin Pattern**
  - Protecting \$
  - Minification Optimizations
  - Context
  - Maintaining Chainability
- **Parameters**
- **Namespacing**
- **Returning Single Values**
- **Maintaining State**

# Naming

## File Name

- `jQuery.myPlugin.js` or `jQuery.myPlugin-1.0.js`
- Include version number inside file
- `jQuery.myPlugin-1.0.min.js`

## Plugin Name

- Don't conflict with any used plugins
- Check name against existing plugins

## Namespace

- One namespace only

# Basic Plugin Pattern

```
(function($){  
    $.fn.myPlugin = function() {  
        return this;  
    };  
})(jQuery);
```

- Immediately-Invoked Function Expression
- Self-Executing anonymous function

# Basic Plugin Pattern Return Value

- `return this;`
- `return this.each();`
- `return this.pushStack();`

# Basic Plugin Pattern

```
(function($){  
    $.fn.myPlugin = function() {  
        return this;  
    };  
})(jQuery);
```

- Immediately-Invoked Function Expression
- Self-Executing anonymous function

# Plugin Parameters

```
.end();
```

```
.css('background-color');
```

```
.css('background-color', 'red');
```

# Our Plugin Parameters

```
.myPlugin(requiredParameter1, optionalParameter1);
```

```
.myPlugin(requiredParameter1, optionalParameter1,  
optionalParameter2, optionalParameter3,  
optionalParameter4)
```



# Plugin Options

```
var options = { parameter1 = 'value1',  
  parameter2 = 'value2',  
  parameter3 = 'value3' };  
  
( 'div' ).myPlugin(options);
```

```
Function(options) {  
  if(options.parameter1 != undefined)  
}
```

# Setting Default Options - Wrong

```
Function(options) {  
    if(options.parameter1 == undefined)  
        // set default  
    if(options.parameter2 == undefined)  
        // set default  
    if(options.parameter3 == undefined)  
        // set default  
    if(options.parameter4 == undefined)  
        // set default  
    if(options.parameter5 == undefined)  
        // set default  
}
```

# Namespaces

# Returning Single Values

- jQuery attr()
- Math plugins
- Content parsing plugins

# Maintaining State

- Use closures to maintain plugin-level state
- Use `.data()` to maintain node-level state

# Summary

- **Internal Techniques**
  - Context of This
  - Parameters
  - Manipulate the Stack
  - Maintain State
- **External Techniques**
  - Naming
  - Chaining
  - Namespacing
  - Closures