# Diving into KnockoutJS

## What is KnockoutJS

### It’s a library, not a framework

#### Ko does not dictate how you structure your code

#### Simply provides plumbing/bindings

#### You are still responsible for writing clean code…

#### It makes writing UI logic fun again…

#### Less error prone as you are not dealing with DOM

#### 14kb, no dependencies

### Model-View-ViewModel

### Class Hierarchy

### Declarative Two-Way Bindings

## Hello World

### Binding Raw Objects

### Binding observables

### Computed observables

### Arrays – foreach & template & context variables

## Application Structure

### Constructor/Object Pattern

### ViewModel Inheritence

### Managing `this`

### Compound ViewModels

### Widgets & `allowBindings: false`

## Working with the server

### Load data asynchronously

### Mapping & ko.toJSON & ko.toJS

## Custom Binding Handlers

### Binding handlers are your friend

#### Should be part of every ko user’s toolbelt. They really are easy to understand…

#### What a binding handler is looking for

#### Utilize existing bindings (simple example… hidden?)

#### Encapsulate logic to write cleaner code (click to edit)

### If you need the DOM in your VM, you are doing it wrong

#### BindingHandlers control \*binding\*. Observables care not.

### Working with other libraries/plugins (jQuery)

#### Markdown Editor

#### jQuery Sortable, AutoComplete, ProgressBar

## Advanced Usage

### Writeable Computed Observables

### Staying DRY with Extensions

### faux dependencies to integrate with other frameworks

### simple editor pattern?