LEARNING semantic-ui

#semantic-ui

Table of Contents

About 1

Chapter 1: Getting started with semantic-ui 2

Remarks 2

Versions 2

Examples 2

Installation via NodeJS - Recommended 3

Chapter 2: ng2-semantic-ui Transitions 4

Syntax 4

Parameters 4

Remarks 4

Examples 4

Simple Scale Transition animation 4

TypeScript Angular 2 Component 4

HTML Partial 5

CSS 5

Credits 7

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: semantic-ui

It is an unofficial and free semantic-ui ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official semantic-ui.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with semantic-ui

Remarks

This section provides an overview of what semantic-ui is, and why a developer might want to use it.

It should also mention any large subjects within semantic-ui, and link out to the related topics. Since the Documentation for semantic-ui is new, you may need to create initial versions of those related topics.

Versions

Version	Release Date
[2.2.10]	2017-03-28
[2.2.9]	2017-02-21
[2.2.8]	2017-02-21
[2.2.7]	2016-12-21
[2.2.6]	2016-10-27
[2.2.5]	2016-10-27
[2.2.4]	2016-08-25
[2.2.3]	2016-08-21
[2.2.2]	2016-07-07
[2.2.1]	2016-06-27
[2.2.0]	2016-07-26
[2.1]	2015-09-02
[1.12]	2015-04-13
[1.0]	2014-11-24
[0.1]	2013-09-25

Examples

Installation via NodeJS - Recommended

Installation via NodeJS is the recommended method. It is preferred because you can use it to build the files selecting just the components you want.

Step 1: Install Node (Link)

Step 2: Install Gulp globally (-g) on your computer

```
npm install -g gulp
```

> Semantic UI uses Gulp to provide command line tools for building themed versions of the library with just the components you need.

Step 3: Install Semantic-UI locally for your project

```
cd /path/to/your/project

npm install semantic-ui --save
```

Step 4: Navigate to the folder where semantic-ui was saved and run the following gulp task (This may be automatically done)

```
cd node_modules/semantic-ui
gulp install
```

This will launch the interactive installer. Just follow the instructions to select the SUI elements that you want and it will build the relevant files for you.

- You will first have to choose the type of installation.First time users can choose the *Automatic Mode* to build SUI
- Then you will have to specify the project folder and where you would like to save SUI (default is `semantic/`)
- In case of Custom installation, you will get the option to pick the components you want.

Step 5: Link to your HTML

Link the compiled CSS and JavaScript file in your HTML

```
<link rel="stylesheet" type="text/css" href="semantic/dist/semantic.min.css">
<script src="semantic/dist/semantic.min.js"></script>
```

All done!

Read Getting started with semantic-ui online: https://riptutorial.com/semantic-ui/topic/7371/getting-started-with-semantic-ui

Chapter 2: ng2-semantic-ui Transitions

Syntax

- **@ViewChild('banner')** is the syntax for gaining access to the DOM element that should be transitioning.
- **varname: SuiTransition** is the syntax
- **new SuiTransition(ElementRef, Renderer)** is the syntax for creating a transition. The `ElementRef` is obtained with `@ViewChild()` and the `Renderer` is obtained through dependency injection. It is a core Angular component.
- **transition.animate(ISuiAnimation)** is the syntax for animating a transition.

Parameters

Parameter	Details
name	type of transition to do
duration	the length of time for the transition to execute

Remarks

This uses `ElementRef` internally which the Angular team says is a security risk.

Examples

Simple Scale Transition animation

TypeScript Angular 2 Component

```
import {Component, OnInit, ViewChild, Renderer} from '@angular/core';
import {SuiTransition} from "ng2-semantic-ui/components/transition/transition";

@Component({
  selector: 'app-home',
  templateUrl: './home.component.html',
  styleUrls: ['./home.component.css']
})
export class HomeComponent implements OnInit {
  @ViewChild('banner') banner;

  myTransition: SuiTransition;

  constructor(private renderer: Renderer) { }

  ngOnInit() {
    this.myTransition = new SuiTransition(this.banner, this.renderer);
    this.myTransition.animate({name: "scale", duration: 1000});
  }

}
```

HTML Partial

```
<div id="home">

  <div class="ui inverted masthead centered segment">
    <div class="ui page grid">

      <!--Should this be in a separate component? Maybe a banner component-->
      <div class="ui hidden information" suiTransition #banner>
        <h1 class="ui inverted centered header">
          Managing information has never been so fun!
        </h1>
        <p class="ui centered lead">
          My software makes it easy to keep track of records.
        </p>
        <a class="large basic inverted animated fade ui button">
          <div class="visible content">Use my software now!</div>
          <div class="hidden content">Go to my software</div>
        </a>
        <div class="ui centered image">
          <img src="../../assets/images/my_background.jpg" alt="My software">
        </div>
      </div>
    </div>


  </div>
</div>
```

CSS

```
#home .masthead {
  background: rgb(24, 42, 115);
  background: -moz-linear-gradient(-45deg, rgba(24, 42, 115, 1) 0%, rgba(33, 138, 174, 1) 69%,
rgba(32, 167, 172, 1) 89%);
  background: -webkit-gradient(linear, left top, right bottom, color-stop(0%, rgba(24, 42,
115, 1)), color-stop(69%, rgba(33, 138, 174, 1)), color-stop(89%, rgba(32, 167, 172, 1)));
  background: -webkit-linear-gradient(-45deg, rgba(24, 42, 115, 1) 0%, rgba(33, 138, 174, 1)
69%, rgba(32, 167, 172, 1) 89%);
  background: -o-linear-gradient(-45deg, rgba(24, 42, 115, 1) 0%, rgba(33, 138, 174, 1) 69%,
rgba(32, 167, 172, 1) 89%);
  background: -ms-linear-gradient(-45deg, rgba(24, 42, 115, 1) 0%, rgba(33, 138, 174, 1) 69%,
rgba(32, 167, 172, 1) 89%);
  background: linear-gradient(135deg, rgba(24, 42, 115, 1) 0%, rgba(33, 138, 174, 1) 69%,
rgba(32, 167, 172, 1) 89%);
  filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#182a73',
endColorstr='#20a7ac', GradientType=1);
  border-radius: 0;
  margin: 0em;
  padding: 1rem 0rem 0;
}

#home .masthead .column {
  position: relative;
}

#home .masthead .information {
  margin: 6em 1em 0 1em;
  text-align: center;
}

#home .masthead .information p {
  display: block;
  text-align: center;
  width: 100%;
  font-weight: 300;
  font-size: 20pt;
}

#home .masthead .information .button {
  margin: 40px auto 20px auto;
  display: block;
  width: 200px;
  border-radius: 500px;
}

#home p.ui.centered.lead {
  font-weight: 300;
  font-size: 16pt;
  padding: 0px 30px;
  line-height: 1.5;
  text-align: center;
  margin-bottom: 0.7em;
}
```

Read ng2-semantic-ui Transitions online: https://riptutorial.com/semantic-ui/topic/7735/ng2-semantic-ui-transitions

Credits

S. No	Chapters	Contributors
1	Getting started with semantic-ui	Community, Romain Vincent, Sarthak, Vinayak Kulkarni
2	ng2-semantic-ui Transitions	Kent Johnson