**Introduction**

Congratulations on taking the plunge!

This AngularJS course is built with the intent of exposing you to the best available resources on each Angular topic. Our desire is to present these topics richly, and from a variety of vantage points, in order to afford you a more complete perspective on them.

# 도입

돌입하시기로 한 것, 축하드립니다!

이 코스의 의도는 여러분에게 AngularJs에 대한 각각의 주제에 대해서 가장 좋은 자료들을 접할 기회를 드리는 것입니다. 우리가 원하는 것은 AngularJS에 대한 주제들을 풍부하게, 그리고 여러 가지 관점에서 다루는 것입니다. 여러분이 이 주제에 대해서 좀 더 완전한 관점을 가질 수 있도록 말이죠.

This course is accompanied by [AngularJS Tutorial: Learn to Build Modern Web Apps with MEAN](https://thinkster.io/angulartutorial/mean-stack-tutorial" \t "_blank).

이 코스는

The learning curve of AngularJS can be described as a hockey stick. Getting started with apps featuring basic functionality is delightfully easy. However, building more complex apps often require understanding Angular's inner workings. Failure to do so will cause development to become awkward and cumbersome.

AngularJs의 학습곡선은 하키채에 비유할 수 있습니다. 기본적인 기능을 사용해서 앱을 만들기 시작하는 것은 즐거울 정도로 쉽죠. 하지만 좀 더 복잡한 앱을 만들려고 하면, 앵귤러의 내부 작동원리에 대한 이해가 필요한 경우가 종종 있습니다. 내부 작동원리를 잘 이해하지 못하면, 복잡한 앱을 만드는 것이 어렵고 느려질 수 있습니다.

With AngularJS, the "Ready, Fire, Aim" learning methodology of duct taping together a handful of tutorials and a cursory glance through the documentation will lead to confusion and frustration. This curriculum is designed to properly guide you through each of the key Angular concepts thoroughly with a broad exposure to high quality content. With your eventual mastery of AngularJS, you will be able to fluently and efficiently construct large-scale applications.

AngularJS의 경우에는, 튜토리얼 몇 개좀 보고 문서를 좀 살펴보고선, "준비, 조준, 발사"와 같은 학습방식이 별로 적절하지 않을 것입니다. 그런 방식으로 접근하면 혼란스럽고 당황스러울 수 있습니다. 이 커리큘럼의 설계 방향은 여러분이 Augnular의 핵심 개념 각각에 대해서 양질의 자료를 살펴보게 함으로서 제대로 안내하는 것입니다. AngularJS를 제대로 마스터한다면, 능숙하게 그리고 효율적으로 큰 규모의 어플리케이션을 개발할 수 있게 될 것입니다.

**Prerequisites**

* Moderate knowledge of HTML, CSS, and JavaScript
* Basic [Model-View-Controller](http://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller) (MVC) concepts
* The [Document Object Model](http://en.wikipedia.org/wiki/Document_Object_Model) (DOM)
* JavaScript [functions](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Functions), [events](https://developer.mozilla.org/en-US/docs/Web/API/Event), and [error handling](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/try...catch)

**Resources**

Since AngularJS is still in its infancy relative to other JavaScript frameworks, the number of encyclopaedic resources on it is still insufficient. Therefore, the curriculum will employ a healthy number of excellent blogs in order to offer a more meaty perspective on respective topics.

* Required Resources
  + [AngularJS - O'Reilly Media](http://shop.oreilly.com/product/0636920028055.do) (available on [Amazon](http://www.amazon.com/AngularJS-Brad-Green/dp/1449344852/ref=sr_1_1?ie=UTF8&qid=1372874049&sr=8-1&keywords=angularjs))
  + [John Lindquist's egghead.io](http://www.egghead.io/)
  + [AngularJS docs](http://docs.angularjs.org/)
* Supplemental Resources
  + [Ben Nadel blog](http://www.bennadel.com/)
  + [OneHungryMind](http://onehungrymind.com/)
  + [year of moo](http://www.yearofmoo.com/)
  + [Bruno Scopelliti blog](http://blog.brunoscopelliti.com/)

**Part 1. Kicking the Tires**

AngularJS is not a library.

Rather, it is a JavaScript framework that embraces extending HTML into a more expressive and readable format. It allows you to decorate your HTML with special markup that synchronizes with your JavaScript leaving you to write your application logic instead of manually updating views. Whether you're looking to augment existing JavaScript applications or harness the full power of the framework to create rich and interactive [SPA's](http://en.wikipedia.org/wiki/Single-page_application), Angular can help you write cleaner and more efficient code.

**Filling the Tank**

We've found that the egghead.io videos are the best starting resource available, so every chapter will lead off with them. The transcribed screencasts and source code are provided along with the videos. We encourage you to follow along with them, as they make the video content much more readily digestible.

As good as the egghead videos are, they should serve only as an introductory resource. Excerpts from the O'Reilly AngularJS book and the angularjs.org documentation complement the videos as the broader and more thorough source, and should be treated as the main reference for the course.

**Adjusting Your Mirrors**

When descending upon an entirely new topic, it is important to frame the topic correctly before diving into the minutia.

Read the following two entries in the AngularJS guide docs, they will give you a good idea of what you're about to get into. Don't worry about picking up on every aspect of the topics they glaze over, all of these will be covered thoroughly in subsequent lessons.



[AngularJS Overview](https://code.angularjs.org/1.2.26/docs/guide/concepts)



[Introduction to AngularJS](https://code.angularjs.org/1.2.26/docs/guide/introduction)

**Revving the Engine**

Before we get on with it, we recommend this post:



[Things I Wish I Were Told About Angular.js](http://ruoyusun.com/2013/05/25/things-i-wish-i-were-told-about-angular-js.html)

It goes over a handful of topics that might be helpful in building the appropriate mental models while consuming the Angular curriculum. Some, probably most, of the terms will bounce right off you until you have gone through that section of the course, but it should provide valuable context when approaching a new topic.

**Now it's off to the races!**