

# Tour of Heroes app and tutorial

## Getting Started

In this tutorial, you build your own application from the ground up, providing experience with the typical development process, as well as an introduction to basic app-design concepts, tools, and terminology.

If you're completely new to Angular, you might want to try the [Try it now](#) quick-start application first. It is based on a ready-made partially-completed project, which you can examine and modify in the StackBlitz interactive development environment, where you can see the results in real time.

The "Try it" tutorial covers the same major topics—components, template syntax, routing, services, and accessing data using HTTP—in a condensed format, following the most current best practices.

This *Tour of Heroes* tutorial shows you how to set up your local development environment and develop an application using the [Angular CLI tool](#), and provides an introduction to the fundamentals of Angular.

The *Tour of Heroes* application that you build helps a staffing agency manage its stable of heroes. The application has many of the features you'd expect to find in any data-driven application. The finished application acquires and displays a list of heroes, edits a selected hero's detail, and navigates among different views of heroic data.

You will find references to and expansions of this application domain in many of the examples used throughout the Angular documentation, but you don't necessarily need to work through this tutorial to understand those examples.

By the end of this tutorial you will be able to do the following:

- Use built-in Angular [directives](#) to show and hide elements and display lists of hero data.
- Create Angular [components](#) to display hero details and show an array of heroes.
- Use one-way [data binding](#) for read-only data.
- Add editable fields to update a model with two-way data binding.
- Bind component methods to user events, like keystrokes and clicks.
- Enable users to select a hero from a master list and edit that hero in the details view.
- Format data with [pipes](#).
- Create a shared [service](#) to assemble the heroes.
- Use [routing](#) to navigate among different views and their components.


You'll learn enough Angular to get started and gain confidence that Angular can do whatever you need it to do.

## Solution

After completing all tutorial steps, the final application will look like this: .


## What you'll build

Here's a visual idea of where this tutorial leads, beginning with the "Dashboard" view and the most heroic heroes:


 Output of heroes dashboard

You can click the two links above the dashboard ("Dashboard" and "Heroes") to navigate between this Dashboard view and a Heroes view.

If you click the dashboard hero "Magneta," the router opens a "Hero Details" view where you can change the hero's name.

 Details of hero in app


Clicking the "Back" button returns you to the Dashboard. Links at the top take you to either of the main views. If you click "Heroes," the application displays the "Heroes" master list view.

 Output of heroes list app


When you click a different hero name, the read-only mini detail beneath the list reflects the new choice.

You can click the "View Details" button to drill into the editable details of the selected hero.

The following diagram captures all of the navigation options.

 View navigations

Here's the application in action:

 Tour of Heroes in Action