**Dependency**

* + [dependencies](https://github.com/npm/npm/blob/2e3776bf5676bc24fec6239a3420f377fe98acde/doc/files/package.json.md#dependencies) are installed on both:
  + npm install from a directory that contains package.json
  + npm install $package on any other directory

Dependencies are installed transitively: if A requires B, and B requires C, then C gets installed, otherwise, B could not work, and neither would A.

Dependencies are required to run only

**Dev Dependency**

[devDependencies](https://github.com/npm/npm/blob/2e3776bf5676bc24fec6239a3420f377fe98acde/doc/files/package.json.md#devdependencies) are:

* also installed on npm install on a directory that contains package.json, unless you pass the --production flag (go upvote [Gayan Charith's answer](https://stackoverflow.com/a/31229205/895245)).
* not installed on npm install "$package" on any other directory, unless you give it the --dev option.
* are not installed transitively.

devDependencies is not installed transitively. E.g. we don't need to test B to test A, so B's testing dependencies can be left out.

 devDependencies only to develop, e.g.: unit tests, CoffeeScript to JavaScript transpilation, minification,

**Local Installation**

Installing it local, means the module will be available only for a project you installed it (the directory you were in, when ran npm install

It simply means that when you install it locally you will not be able to use any directory to create the project instead you will have to create the project within the installation directory.

**Global Installation**

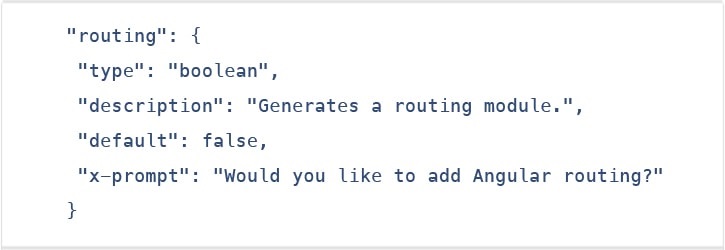
Global install, instead puts the module into your Node.js path (OS dependent), and will be accessible from any project, without the need to install it separately for each.

It simply means that when you install it globally you will be able to use any directory of your choice and you will be able to create project anywhere rather than of the installation directory.

Angular 7 Features

**1. CLI prompts**

In Angular 7, the CLI prompts have been updated to v7.0.2 with new features. For instance, it will now prompt users when typing commands like @angular/material, ng-new, and ng-add to help them discover the in-built SCSS support, routing, and more.



These CLI prompts, in addition, have been added to Schematics, so that all package publishing schematics can now benefit from CLI prompts.

### 2. Angular material & component dev kit (CDK)

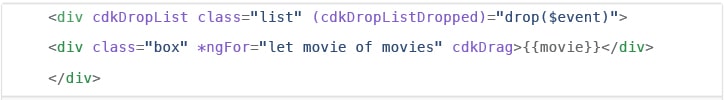
The Angular 7 introduced minor visual updates & improvements in Material Design that earlier received a [major update](https://www.youtube.com/watch?v=1Dh8ZBQp9jo) this year only.

In addition, refresh, virtual scrolling, large lists of data, dynamic loading and unloading of parts of the DOM also were the part of improvements in CDK and Angular Material. Furthermore, the applications in Angular 7 can now also be fitted with drag-and-drop functionality by either importing ScrollingModule or DragDropModule.

### 3. Drag & drop

The new drag-drop module basically provides a better way to easily create drag & drop interfaces, which is backed by sorting within a list, support for free dragging, animations, custom drag handles, transferring items between lists, previews, and placeholders.

In simple terms, the Drag-and-Drop support has now been implemented in CDK and it also includes automatic rendering as the users relocates items.





### 4. Virtual scrolling

Like mentioned earlier, the new Virtual Scrolling in Angular 7 basically loads and unloads items from the DOM depending upon visible parts of lists, resulting into a much faster experiences for users having huge scrollable lists.

This virtual scrolling package basically provides helpers, which react to all scroll events.



Simply put, it activates a high-performant way by making the height of container element exactly same as the height of total number of remaining elements to be rendered.

This, in turn, then renders the only visible items in view, creating faster experiences for the end-users.

### 5. Application performance improvements

The [development team](https://www.peerbits.com/our-team.html) at Google have always focused on the performance improvements, and while doing so, they recently found that most of the developers were using reflect-metadata in their production, which actually was only required in the development.



So, to fix this problem, they’ve made a part of Angular JS 7 to automatically remove this from the polyfills.ts file.

### 6. No Ivy

No Ivy rendered in the Angular 7 according to the official information. The [AngularJS development company](https://www.peerbits.com/hire-angularjs-developers.html)’s team have said that the Ivy is in the pipeline; however, they haven’t disclosed its final timeline.

The official blog post also mentioned that the backward compatibility validation has begun. And its full beta version is expected to launch with Angular 8 version.

### 7. Documentation updates

Another key improvement introduced in the Angular 7 is of the documentation update.

The team has worked hard on improving the reference material and the guidelines for the betterment and convenience of the developers.

The documentation updates for Angular is an important step for the Angular CLI.

### 8. Dependency updates

Documentation is not the only things that have been updated. Even the dependencies have undergone upgradation for the third-party projects.

The support for Node 10, TypeScript3.1, and the RxJS6.3 all are included under this update.

However, you would continue to receive the support if you already have Node 8. Talking about the latest update of TypeScript 3.1, it has now become compulsory for Angular 7 users to bump to TS 3.1