**Lesson31 Pre Loading Angular Modules**

**Notes:-**

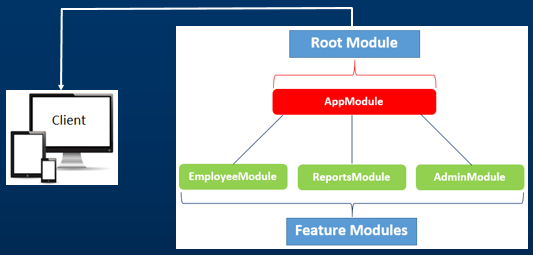
**1-There are 3 types of Module Loading strategies in angular**

**A-Eager Loading**

**B-Lazy Loading**

**C-Pre Loading**

**A-Eager Loading: - all the modules must be downloading to the client machine before the application starts**



**1-By Default all the modules are eagerly loaded**

**2-The root module is always eagerly loaded**

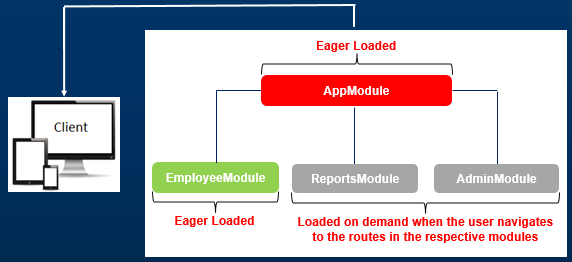
**3-eager loading works fine for small applications**

**4-only the first request to the application takes long times, But the subsequent requests from the same client will be faster**

**5-to make feature module to be Eager, it must referenced from the**

**App.module.ts**

**B-Lazy loading: Lazy loaded modules are loaded on demand when the user navigates to the routes in those respective modules.**



**1-Lazy loaded modules are loaded on demand when the user navigate to the routes in those respective modules.**

**2-To lazy load a module, it should not be referenced in any other modules.**

**3-Lazy loading can significantly reduce the application load time.**

**4-Downside of lazy loading, when a route in a lazy loaded module is first requested, the user has to wait for that module to be downloaded**

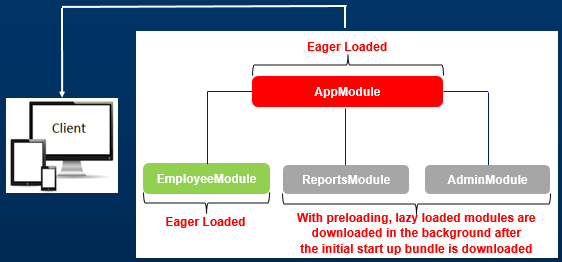
**Question: why should the application wait to download a lazy loaded module until we navigate to a route in that module**?

**Can the application download the lazy loaded module in the background after the initial bundle that is required to start the application is downloaded?**

**Well, yes, the lazy loaded modules can be preloaded in the background after the initial startup bundle is downloaded. Here is how it works.**

**Preloading in Angular:**

**Preloading is the same as lazy loading but happens slightly differently.**



**1-First, the module to bootstrap the application and eager loaded modules are downloaded.**

**2-At this point, we have the application up and running and the user is interacting with the application.**

**3-While the application has nothing else to download, it downloads angular modules configured to preload in the background.**

**4-So, by the time the user navigates to a route in a lazy loaded module, it is already pre-loaded, so the user does not have to wait, and sees the component associated with that route right away.**

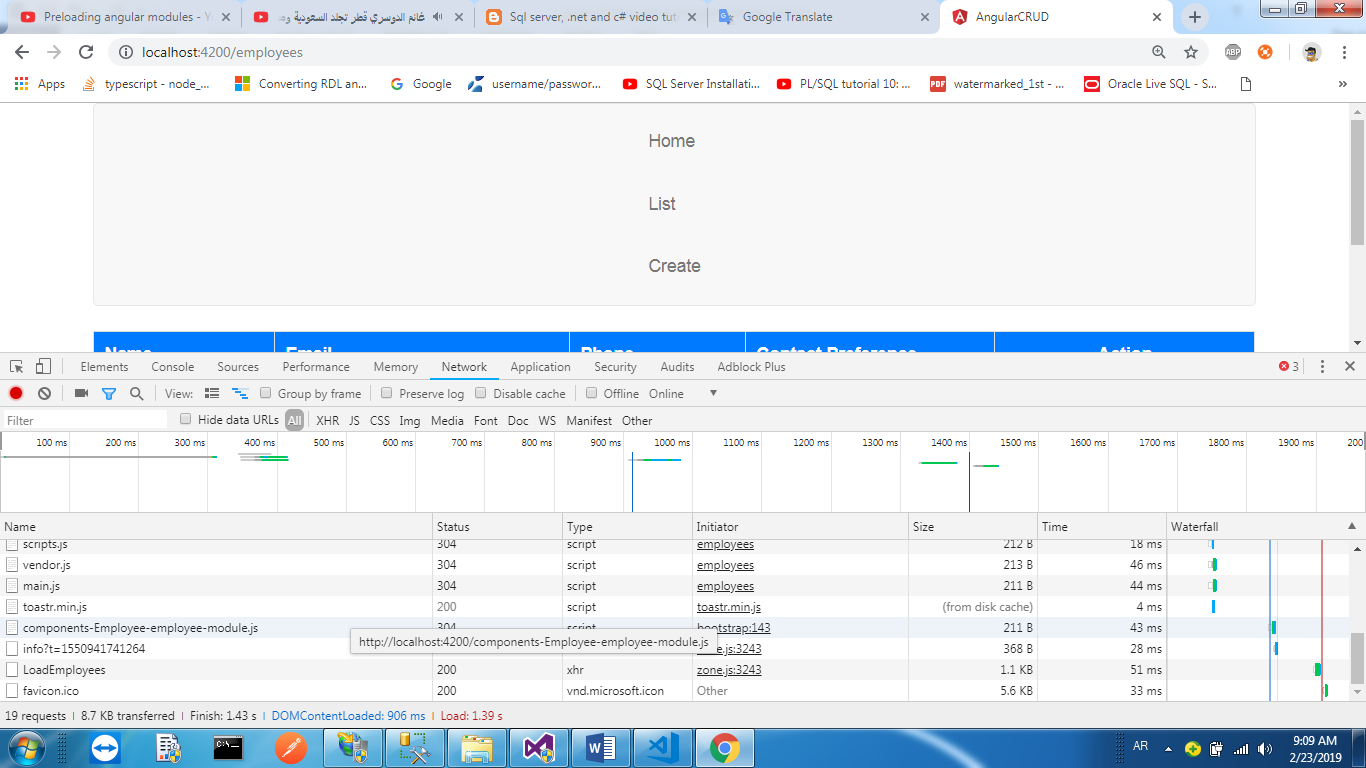
**5-So with preloading modules, we have the best of both the worlds’ i.e Eager Loading and Lazy Loading.**

**6-Preloading is also often called Eager Lazy Loading**

**Example:-**

**1-when loading the page, we see that the feature module it downloaded at the first time.**

**2-when you want to navigate to another page, we see its load directly without need to download from the server**



**The value for preloadingStrategy property can be one of the following**

|  |  |
| --- | --- |
| **Value** | **Description** |
| NoPreloading | This is the default and does not preload any modules  (set it as Lazy loading) |
| PreloadAllModules | Preloads all modules as quickly as possible in the background |
| Custom Preload Strategy | We can also specify our own custom preloading strategy. We will discuss why and how to implement custom preloading strategy in our next video. |