**Route Guards**

Route Guards are like **security guards** in your Angular app.  
They **control who can enter or leave** a particular page (route).

Imagine your Angular app is a **building with many rooms** 🏢  
Each **page (route)** is a **room**, and **Route Guards** are the **security guards at each door** 🚪

They check:

* "Can you go inside this room?"
* "Can you leave this room?"
* "Should we load this room at all?"
* "Do we have all data ready before entering?"

Let’s say you have two pages:

* /login
* /dashboard

If a user is **not logged in**, they **should not be able to go to /dashboard**.

That’s where a **Route Guard** helps!

| **Guard Type** | **What It Does** | **Real-life Example** |
| --- | --- | --- |
| **CanActivate** | Stops users from entering a route if they don’t have permission | Only logged-in users can open Dashboard |
| **CanDeactivate** | Stops users from leaving a route if something is not saved | Show “Do you want to leave without saving?” when leaving a form |
| **Resolve** | Loads data *before* showing the route | Fetch student details before showing the profile page |
| **CanLoad** | Stops a lazy-loaded module from loading | Don’t even load the admin module if the user is not admin |

**🧭 1 Normal Routing (Eager Loading)**

🟢 **Definition:**  
All modules (and their components) load **as soon as the app starts** — even if you don’t visit that page.

🧩 **Example (like your earlier routing example):**

const routes: Routes = [

{ path: '', component: HomeComponent },

{ path: 'student', component: StudentComponent },

];

✅ **Meaning:**  
When you run the app, **both Home and Student components** load into memory immediately.  
So it’s simple but can make the app **slower** if there are many pages.

**🚀 2️ Lazy Loading Routing**

🟡 **Definition:**  
Modules load **only when needed** (only when you visit that route).

🧩 **Example (from today’s lazy loading example):**

const routes: Routes = [

{ path: '', component: HomeComponent },

{

path: 'student',

loadChildren: () => import('./student/student.module').then(m => m.StudentModule)

}

];

✅ **Meaning:**  
When you first open the app, **only Home** loads.  
The **StudentModule** loads **later** (only when user clicks “Go to Student Page”).  
This helps the app **start faster** and **saves memory**.

**Note : ng g m student --route student --module app.module**