



# **LAZY LOADING**



# WHAT IS LAZY LOADING?

- Lazy loading is a design pattern that delays the loading of modules or components until they are required
- Instead of loading all modules upfront, Angular loads them on-demand, which reduces the initial bundle size and improves load time



# HOW TO IMPLEMENT LAZY LOADING IN ANGULAR



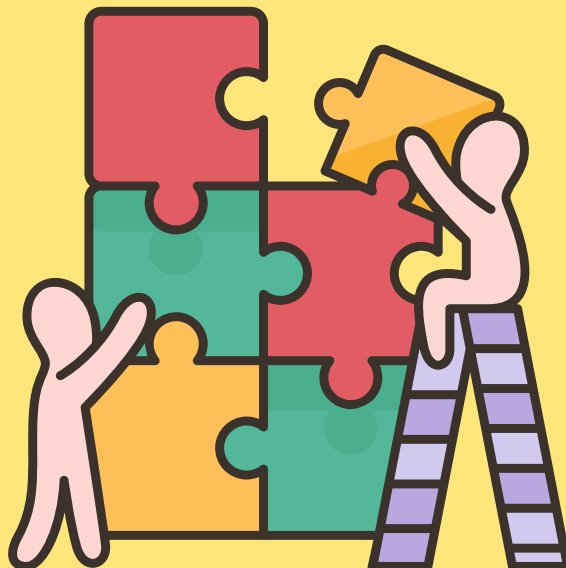
## STEP 1: CREATE A FEATURE MODULE

- First, create a feature module that you want to load lazily
- let's create a module named ReportsModule.



BASH

```
ng generate module reports --route reports
```



## THIS COMMAND DOES A FEW THINGS

- Creates the ReportsModule
- Adds a route for lazy loading (--route reports)
- Registers it in the AppModule (--module app.module)



## STEP 2: SET UP ROUTING FOR LAZY LOADING

- Angular automatically adds the lazy loading route in the AppRoutingModuleModule

```
const routes: Routes = [
  { path: 'reports', loadChildren: ()
    => import('./reports/reports.module')
      .then(m => m.ReportsModule) }
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule { }
```



## STEP 3: CREATE COMPONENTS WITHIN THE FEATURE MODULE

- Inside the ReportsModule, you can create components specific to this module

```
BASH  
  
ng generate component reports/report-list
```





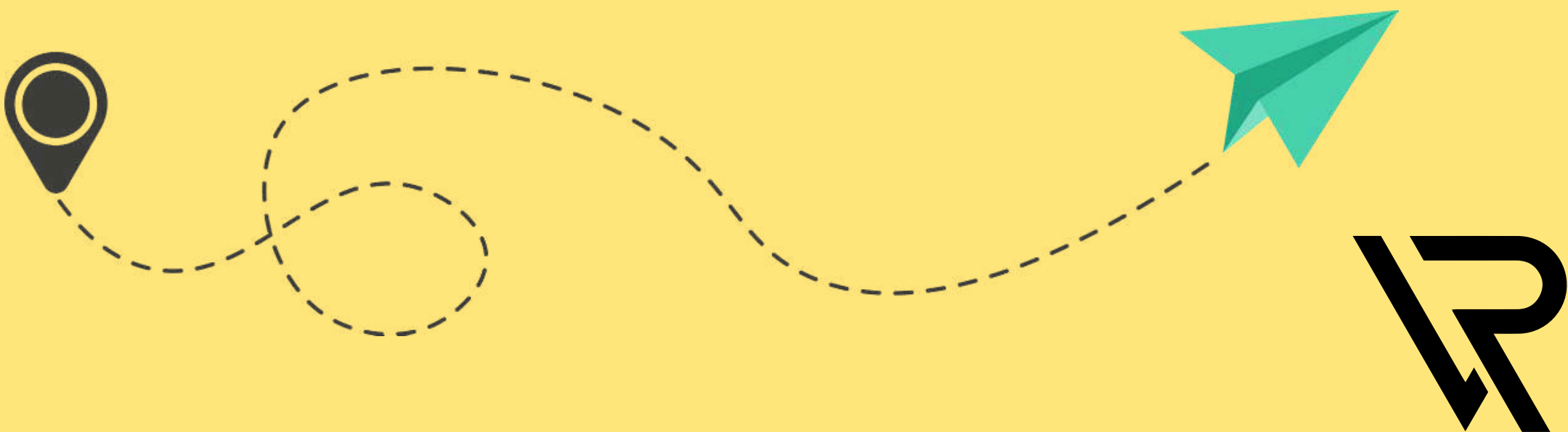
## STEP 3: CREATE COMPONENTS WITHIN THE FEATURE MODULE

- Update the ReportsRoutingModule with the routes for the components

```
TYPESCRIPT

const routes: Routes = [
  { path: '', component: ReportListComponent }
];

@NgModule({
  imports: [RouterModule.forChild(routes)],
  exports: [RouterModule]
})
export class ReportsRoutingModule { }
```



## STEP 4: TEST YOUR LAZY LOADED MODULE

- Run your application and navigate to the /reports route
- You'll notice that the ReportsModule is only loaded when this route is accessed, resulting in faster initial load times





# Did you find it **Useful?**

Leave a **comment!**



**Alamin CodePapa**

@CodePapa360

**FOLLOW FOR MORE**

Like



Comment



Repost

