**Creación de un nuevo módulo para uso de lazy load en Angular**

**Objetivo**: Explicar el proceso de creación de un nuevo módulo en una aplicación Angular utilizando lazy load, con el fin de mejorar el rendimiento y la carga de la página.

**Recursos necesarios:**

* Conocimientos básicos de desarrollo web con Angular.
* Un entorno de desarrollo Angular configurado.

**Pasos:**

**1. Creación del nuevo módulo:**

* En la terminal, navegar hasta la raíz de tu proyecto Angular.
* Ejecutar el siguiente comando para generar un nuevo módulo:

ng generate module nombre-modulo --route ruta-modulo --module modulo-padre

Sustituir "nombre-modulo" por el nombre deseado para el módulo, "ruta-modulo" por la ruta relativa que se utilizará para acceder al módulo y "modulo-padre" por el nombre del módulo padre en el que se incluirá el módulo creado.

* El comando generará una nueva carpeta con el nombre del módulo y los archivos necesarios.

**2. Configuración de la ruta lazy load:**

* Abrir el archivo de enrutamiento del módulo padre (generalmente "app-routing.module.ts").
* Importar el módulo generado en el paso anterior utilizando la siguiente sintaxis:

const routes: Routes = [

// Otras rutas existentes

{

path: 'ruta-modulo',

loadChildren: () => import('./ruta-modulo/nombre-modulo.module').then(m => m.NombreModuloModule)

}

];

Asegurarse de reemplazar "ruta-modulo" por la ruta definida en el comando anterior y "nombre-modulo" por el nombre del módulo generado.

**3. Implementación del módulo:**

* Abrir el archivo de módulo generado (generalmente "nombre-modulo.module.ts").
* Definir y configurar los componentes, servicios y otros recursos necesarios dentro del módulo.
* Exportar el módulo utilizando la siguiente sintaxis:

@NgModule({

declarations: [Componente1, Componente2],

imports: [CommonModule, FormsModule, ReactiveFormsModule],

providers: [Servicio1, Servicio2]

})

export class NombreModuloModule { }

**4. Uso del módulo en la aplicación:**

* En el módulo padre, importar y agregar el módulo al arreglo de imports:

imports: [

// Otros módulos importados

NombreModuloModule

]

**5. Prueba de lazy load:**

* Iniciar la aplicación Angular y navegar a la ruta definida para el nuevo módulo.
* Comprobar que el módulo se carga de forma diferida cuando se accede a su ruta correspondiente.

Con estos pasos puedes crear un nuevo módulo utilizando lazy load, lo que permitirá una carga más eficiente y mejorar el rendimiento general.