

## Prerendering static pages

Angular Universal lets you prerender the pages of your application. Prerendering is the process where a dynamic page is processed at build time generating static HTML.

### How to prerender a page

To prerender a static page make sure to add SSR capabilities to your application. For more information see the universal guide. Once SSR is added, run the following command:

```
npm run prerender
```

### Build options for prerendering

When you add prerendering to your application, the following build options are available:

- **browserTarget:** Specify the target to build.
- **serverTarget:** Specify the Server target to use for prerendering the application.
- **routes:** Define an array of additional routes to prerender.
- **guessRoutes:** Whether builder should extract routes and guess which paths to render. Defaults to **true**.
- **routesFile:** Specify a file that contains a list of all routes to prerender, separated by newlines. This option is useful if you have a large number of routes.
- **numProcesses:** Specify the number of CPUs to be used while running the prerendering command.

### Prerendering dynamic routes

You can prerender dynamic routes. An example of a dynamic route is `product/:id`, where `id` is dynamically provided.

To prerender dynamic routes, choose one from the following options: \* Provide additional routes in the command line \* Provide routes using a file \* Prerender specific routes

**Provide additional routes in the command line** While running the prerender command, you can provide additional routes. For example:

```
ng run <app-name>:prerender --routes /product/1 /product/2
```

**Providing additional routes using a file** You can provide routes using a file to generate static pages. This method is useful if you have a large number of

routes to generate, such as product details for an e-commerce application, which might come from an external source (Database or CMS).

To provide routes using a file, use the `--routes-file` option with the name of a `.txt` file containing the routes.

For example, you could generate this file by using a script to extract IDs from a database and save them to a `routes.txt` file:

```
/products/1 /products/555
```

When your `.txt` file is ready, run the following command to prerender the static files with dynamic values:

```
ng run <app-name>:prerender --routes-file routes.txt
```

**Prerendering specific routes** You can also pass specific routes to the prerender command. If you choose this option, make sure to disable the `guessRoutes` option.

```
ng run <app-name>:prerender --no-guess-routes --routes /product/1 /product/1
```