



Table of Contents >

# .env File

## At Build, Dev, and Generate Time

Nuxt CLI has built-in dotenv support in development mode and when running `nuxi build` and `nuxi generate`.

In addition to any process environment variables, if you have a `.env` file in your project root directory, it will be automatically loaded **at build, dev, and generate time**, and any environment variables set there will be accessible within your `nuxt.config` file and modules.

If you want to use a different file – for example, to use `.env.local` or `.env.production` – you can do so by passing the `--dotenv` flag when using `nuxi`. For example:

```
> npx nuxi dev --dotenv .env.local
```

Just as above, this will only apply in development mode and when running `nuxi build` and `nuxi generate`.

When updating `.env` in development mode, the Nuxt instance is automatically restarted to apply new values to the `process.env`.

Note that removing a variable from `.env` or removing the `.env` file entirely will not unset values that have already been set.

## Production Preview

**After your server is built**, you are responsible for setting environment variables when you run the server. Your `.env` file will not be read at this point. How you do this is different for every environment.

For local production preview purpose, we recommend using `nuxi preview` since using this command, the `.env` file will be loaded into `process.env` for convenience. Note that this command requires dependencies to be installed in the package directory.

Or you could pass the environment variables as arguments using the terminal. For example, on Linux or macOS:


```
> DATABASE_HOST=mydatabaseconnectionstring node .output/server/index.mjs
```

Note that for a purely static site, it is not possible to set runtime configuration `config` after your project is prerendered.

 [Read more in Docs > Guide > Going Further > Runtime Config.](#)

If you want to use environment variables set at build time but do not care about updating these down the line (or only need to update them reactively *within* your app) then `appConfig` may be a better choice. You can define `appConfig` both within your `nuxt.config` (using environment variables) and also within an `~/app.config.ts` file in your project.

 [Read more in Docs > Guide > Directory Structure > App Config.](#)

 [Edit on Github](#)



© 2016-2023 Nuxt - MIT  
License

Enterprise

Design Kit  
Nuxt Studio

NuxtLabs

