

## Overview

Stackblitz is an online tool for creating, collaborating and sharing ideas. In AIO we use it to share one or more runnable versions of our examples.

Stackblitz can be used both on a separate page and in an embedded form. \*  
`generateStackblitz.mjs` - executes each of the `StackblitzBuilder` to generate a stackblitz file for each example.

## Stackblitz generation

Both forms are created within `builder.mjs`. How is a stackblitz created? What is the process from a directory with files to a link with a stackblitz.

An “executable” stackblitz is an HTML file with a `<form>` that makes a post to stackblitz on submit. It contains an `<input>` element for each file we need in the stackblitz.

The form will be submitted on load, so you can either double click the HTML file or open it with an anchor tag to open the stackblitz.

So the `builder.mjs` job is to get all the needed files from an example and build this HTML file for you.

## Customizing the generation per example basis

How does this tool know what is an example and what is not? It will look for all folders containing a `stackblitz.json` file. If found, all files within the folder and subfolders will be used in the stackblitz, with a few generic exceptions that you can find at `builder.mjs`.

You can use the `stackblitz.json` to customize the stackblitz generation. For example:

```
{
  "description": "Tour of Heroes: Part 6",
  "files": [
    "!**/*.d.ts",
    "!**/*.js",
    "!**/*. [1,2].*"
  ],
  "tags": ["tutorial", "tour", "heroes", "http"]
}
```

Here you can specify a description for the stackblitz, some tags and also a files array where you can specify extra files to add or to ignore.

## Executing the stackblitz generation

`generateStackblitz.mjs` will create a stackblitz for each `stackblitz.json` it finds.

Where? At `src/generated/live-examples/`.

Then the `<live-example>` embedded component will look at this folder to get the stackblitz it needs for the example.

## Appendix: Why not generating stackblitz at runtime?

At AngularJS, all the plunker examples were generated at runtime. The downside was that all the example code had to be deployed as well and would no longer be useful after the plunker was generated.

This `StackblitzBuilder` tool takes a few seconds to run, and the end result is only 3mb~.