



Creating a WordPress plugin with multiple blocks

Most of the WordPress block plugins you see online and most of the examples showing you how to create a block plugin for WordPress show a single block per plugin.

This may not always be what we want to do.

When creating blocks for a client or creating a set of blocks for a theme I would rather provide a single plugin with all necessary blocks rather than provide one plugin per block as most of the tools and examples provide.

I will follow Ryan Welcher's method to add multiple blocks per plugins as shown in the video below:

The process leverages create-blocks ability to create templates for other people to use.

Ryan Welcher has created a [template](#) to work with multiple blocks in a single plugin.

```
npx @wordpress/create-block example-plugin \
```

```
--template @ryanwelcher/multiple-blocks-template
```

As Ryan indicates in the video, there's still some manual work to do to get the block working properly but it's significantly less than you would have to do if starting from scratch.

The main change is adding a `webpack.config.js` configuration file at the root of the plugin directory.

In this file we do the following:

We first require the default Webpack configuratin from `@wordpress/scripts` and merge it with a new entry declaration pointing to the blocks we want to add, one entry per block in the package.

```
const defaultConfig = require( '@wordpress/scripts/config/webpack.config'

module.exports = {
  ...defaultConfig,
  entry: {
    'block-one': './includes/block-editor/blocks/block-one',
  },
};
```

The work is further reduced in version 4.0 of the [@wordpress/create-block](#) package (the latest at the time of writing).

This version introduces the `--no-plugin` flag to make it easier to create additional blocks inside an existing plugin

The idea is that running the `@wordpress/scripts` tool with a template Ryan created, we can prepare the plugin to handle multiple blocks and then further reduce the manual edits by using `@wordpress/create-blocks` to create the block inside the structure of the plugin using the `--no-plugin` flag to only create the block.

Then update `wepback.config.js` with the information about the new block.

This