

# Hiding blocks from the block inserter

Hiding blocks allow the blocks to be used for existing content, but not for new content.

```
function get_plugin_settings() {
    $disabled_blocks = [];
    $disabled_blocks[] = 'my/blockname';

    return [
        ['disabledBlocks' => $disabled_blocks,
        ];
    }

    # enqueue scripts hook:
    wp_localize_script(
        'myblocks-editor',
        'myBlocksSettings',
        get_plugin_settings()
    );
}
```

Javascript:

```
import { addFilter } from '@wordpress/hooks';

// Define disabled blocks.
// If you need to rely on logic available only in PHP,
// pass this data using a global variable instead.
const DISABLED_BLOCKS = window.myBlocksSettings.disabledBlocks;

/**
 * Conditionally enable/disable insertion of blocks.
 *
 * Ensure sure this function runs BEFORE you register your blocks.
 *
 * @param {object} settings block type definition
 * @param {string} name name of the block type
 */
```

```
* @returns {object} block type settings
*/
function filterBlockRegistration( settings, name ) {
  if ( ! DISABLED_BLOCKS.includes( name ) ) {
    return settings;
  }

  // Ensure there is a supports section
  if ( undefined === settings.supports ) {
    settings.supports = {};
  }

  // Disable the UI to add this block .
  // If the block is added in another way,
  // e.g. legacy, programmatically, copy-paste, it will still work.
  settings.supports inserter = false;

  return settings;
}

addFilter(
  'blocks.registerBlockType',
  'myNamespace',
  filterBlockRegistration
);
```

---

Revision #3

Created 2022-10-27 09:52:33 UTC by Sander

Updated 2022-10-27 11:01:23 UTC by Sander