private function cleanDir($dir)

{

$iterator = new \DirectoryIterator(\Yii::getAlias($dir)); foreach($iterator as $sub) {

if(!$sub->isDot() && $sub->isDir()) {

$this->stdout('Removed ' . $sub->getPathname() . PHP\_EOL);

FileHelper::removeDirectory($sub->getPathname());

}

}

}

}

2. Now we can use our own console controller with default settings. Just run the yii shell script: . /yii

3. Look for own clean commands:

This is Yii version 2.0.7.

The following commands are available:

- asset Allows you to combine...

asset/compress Combines and compresses the asset.

asset/template Creates template of configuration file.

- clean

clean/assets

clean/runtime

Removes content of assets and runtime directories. Removes temporary assets.

Removes runtime content.

- fixture Manages fixture data loading and unloading.

fixture/load (default) Loads the specified fixture data. fixture/unload Unloads the specified fixtures.

4. Right now run asset cleaning:

.yii clean/assets

5. See the process report:

Removed /yii-book.app/web/assets/25f82b8a Removed /yii-book.app/web/assets/9b3b2888 Removed /yii-book.app/web/assets/f4307424 Done

6. If you want to use this controller in the yii2 -app- advanced application, just specify the custom working paths:

return [

'id' => 'app-console',

'basePath' => dirname( DIR ),

'bootstrap' => ['log'],

'controllerNamespace' => 'console\controllers',

'controllerMap' => [

'clean' => [

'class' => 'console\controllers\CleanController',

'assetPaths' => [

'@backend/web/assets',

'@frontend/web/assets',