<?= \yii\grid\GridView::widget([

'dataProvider' => $dataProvider,

'columns' => [

'id',

'created\_at:datetime',

'title',

'value:number',

],

]) ?>

4. Also, you can extend every existing component without overwriting its source code.

How it works...

To be able to attach a component to an application, it can be extended from the yii\base\component class. Attaching is as simple as adding a new array to the components section of configuration. There, a class value specifies the component’s class, and all other values are set to a component through the corresponding component’s public properties and setter methods.

Implementation itself is very straightforward; we are wrapping the <http://api.fixer.io>calls into a comfortable API with validators and caching. We can access our class by its component name using Yii : : $app. In our case, it will be Yii: :$app->exchange.

See also

[For official information about components, refer to http://www.yiiframework.com/doc-2.0/guide-concept-](http://www.yiiframework.com/doc-2.0/guide-concept-components.html) components.html.

For the NumberHelper class sources, refer to the Creating helpers recipe.