* delete /films/5: This deletes film 5
* options /films: This shows the supported verbs regarding the /films endpoint
* options /films/5: This shows the supported verbs regarding the /films/5 endpoint

It works like this because \yii\rest\ActiveController supports the following actions:

* index: This lists the models
* view: This returns the details of a model
* create: This creates a new model
* update: This updates an existing model
* delete: This deletes an existing model
* options: This returns the allowe d HTTP methods

And there’s also a verbs() method that defines the allowed request methods for each action.

To check that our RESTful API is working correctly, let’s send several requests.

Let’s begin with the get request. Run this in the console:

**curl -i -H "Accept:application/json" "<http://yii-book.app/films>"**

You will get the following output:

HTTP/1.1 200 OK

Date: Wed, 23 Sep 2015 17:46:35 GMT Server: Apache X-Powered-By: PHP/5.5.23 X-Pagination-Total-Count: 5 X-Pagination-Page-Count: 1 X-Pagination-Current-Page: 1 X-Pagination-Per-Page: 20

Link: <<http://yii-book.app/films?page=1>>; rel=self Content-Length: 301

Content-Type: application/json; charset=UTF-8

[{"id":1,"title":"Interstellar","release\_year":2014},{"id":2,"title":"Harry Potter and the Philosopher's Stone","release\_year":2001},{"id":3,"title":"Back to the Future","release\_year":1985},{"id":4,"title":"Blade Runner","release\_year":1982}, {"id":5,"title":"Dallas Buyers Club","release\_year":2013}]

Let’s send a post request. Run this in the console:

**curl -i -H "Accept:application/json" -X POST -d title="New film" -d release\_year=2015 "<http://yii-book.app/films>"**

You will get the following output:

HTTP/1.1 201 Created

Date: Wed, 23 Sep 2015 17:48:06 GMT

Server: Apache

X-Powered-By: PHP/5.5.23

Location: http: //yii-book.app/films/6

Content-Length: 49

Content-Type: application/json; charset=UTF-8 {"title":"New film","release\_year":"2015","id":6}