



Nakopnite webovú aplikáciu!

@RoboNovotny | Bezadis

web?

JavaScript

Java

Python

PHP

.NET

.NET

PHP

Ruby

web?

Spring

Java EE

HATERS GONNA HATE



1. Java + XML

2. build tool

3. WAR

3. application server

backend

...makes it easy to
create stand-alone,
production-grade
Spring based
Applications that
you can "just run"

Spring Boot

```
curl https://start.spring.io/install.sh | sh
```



```
@RestController  
class Hello {  
    @RequestMapping("/")  
    String home() {  
        "Hello World!"  
    }  
}
```

```
@RestController  
class Hello {  
    @RequestMapping("/")  
    String home() {  
        "Hello World!"  
    }  
}
```

```
spring run app.groovy
```

spring init -x

<http://start.spring.io/>

```
spring init -d=web,actuator,hsqldb,jpa -x
```

actuator amqp aop batch cloud-
connectors elasticsearch facebook
freemarker gemfire groovy-
templates hateoas hornetq
integration jdbc jersey jetty jpa jta-
atomikos jta-bitronix linkedin log4j
log4j2 logging mail mobile
mongodb mustache parent redis
remote-shell rest security solr test
thymeleaf tomcat twitter undertow
validation velocity web websocket
ws

/health

/metrics

/trace

```
spring.datasource.url=jdbc:mysql://localhost/test
```

```
spring.datasource.username=dbuser
```

```
spring.datasource.password=dbpass
```

```
pom.xml > jdbc-starter + mysql-jdbc
```

```
application.properties
```


front

end





bower

```
npm install -g bower
```

node.js

```
{  
  "name": "hateomat"  
}
```

bower.json

```
{  
  "directory":  
    "src/main/resources/static/bower_components"  
}
```

bower.json



bootstrap

```
bower install bootstrap --save
```

bower.json


```
npm install wiredep
```

node

```
require('wiredep')({  
  src:  
    [  
      'src/main/resources/static/index.html'  
    ]  
});
```

wiredep

```
<!-- bower:css -->
```

```
<!-- endbower -->
```

```
<!-- bower:js -->
```

```
<!-- endbower -->
```

wiredep



```
bower install angular --save
```

bower.json

```
bower install angular-resource --save
```

bower.json

JS
/
HTML

Scope

Controller

Service

```
<body ng-app="hateomat">
```

```
<div ng-controller="HateController">  
  <div ng-repeat="hate in hates">  
    {{ hate.description }}  
  </div>  
</div>
```

angular


```
var HateController = function($scope ...) {  
  $scope.hates = [...]  
  
  $scope.newHate = {};  
  
  $scope.addHate = function() {  
    ...  
  }  
};
```

```
<form ng-submit="addHate()">
```

```
<input type="text" id="description"  
      ng-model="hate.description"/>
```

```
<input type="text" id="author"  
      ng-model="hate.author"/>
```

```
<button type="submit">Submit</button>  
</form>
```

angular

```
angular
```

```
.module("hateomat", [])
```

```
.controller('HateController', HateController)
```

angular

```
var HateController =  
    function($scope, HateService) {  
    }  
};
```

```
HateController.$inject = ['$scope', 'HateService'];
```

angular

```
var HateController =  
    function($scope, HateService) {  
    }  
};
```

```
HateController.$inject = ['$scope', 'HateService'];
```

angular

angular

```
.module("hateomat", ['ngResource'])
```

```
.controller('HateController', HateController)
```

ngResource

```
var HateService = function($resource) {  
  return $resource("/hates");  
}  
CityService.$inject = ['$resource'];
```

ngResource

angular

```
.module("hateomat", [])
```

```
.controller('HateController', HateController)
```

```
.service('HateService', HateService)
```

service


```
$scope.newHate = {};
```

```
HateService.query(function (response) {  
  $scope.hates = response;  
});
```

```
$scope.addCity = function() {  
  HateSService  
    .save($scope.newHate, function(newHate) {  
      $scope.hates.push(newHate);  
      $scope.newHate = {};  
    })  
}
```

controller

demo

<http://bit.ly/1zks3ty>

