

Learning Languages With Wikipedia

Content

- Several parts...
 - Parser
 - WebApp
 - User Interface
 - Statistics
 - Junit Tests
- ...one program

Parser

- Initialisierung
 - Read in supported languages
 - Read in lookup tables
- Parsing with Jsoup (1)
 - Read paragraphs
 - Tokenize

Parser

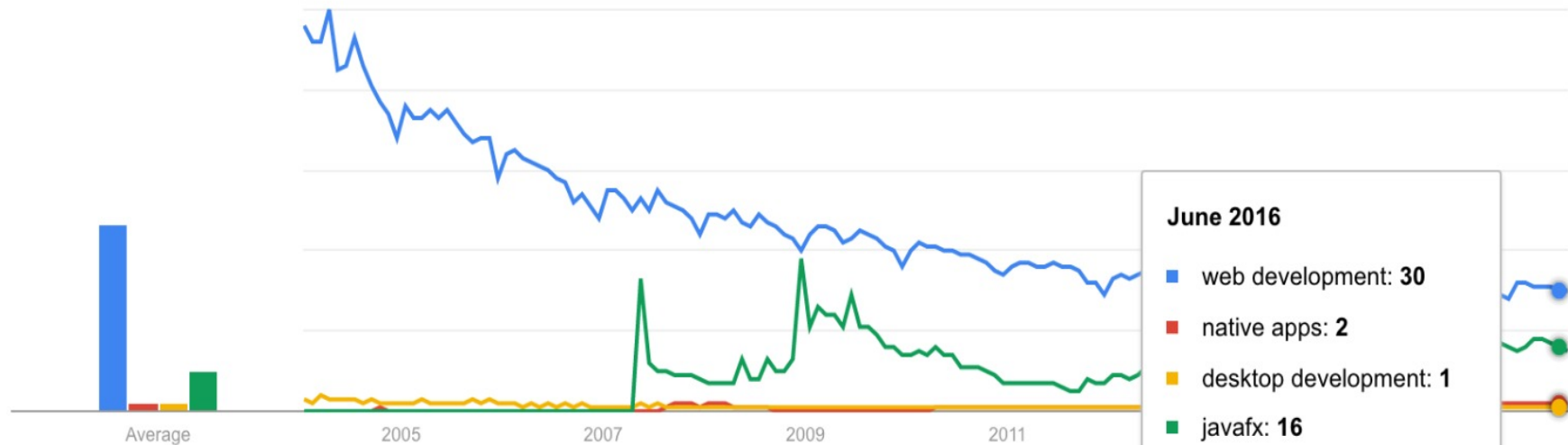
- Parsing with Jsoup(2)
 - Remove token, mark and number position
 - Random
 - Lookup table
 - Generate wordlist
 - From removed token
 - From Lookup table
 - Replace marker with respective dropdown wordlist
- Pass back changed html

WebApp

Interest over time ?

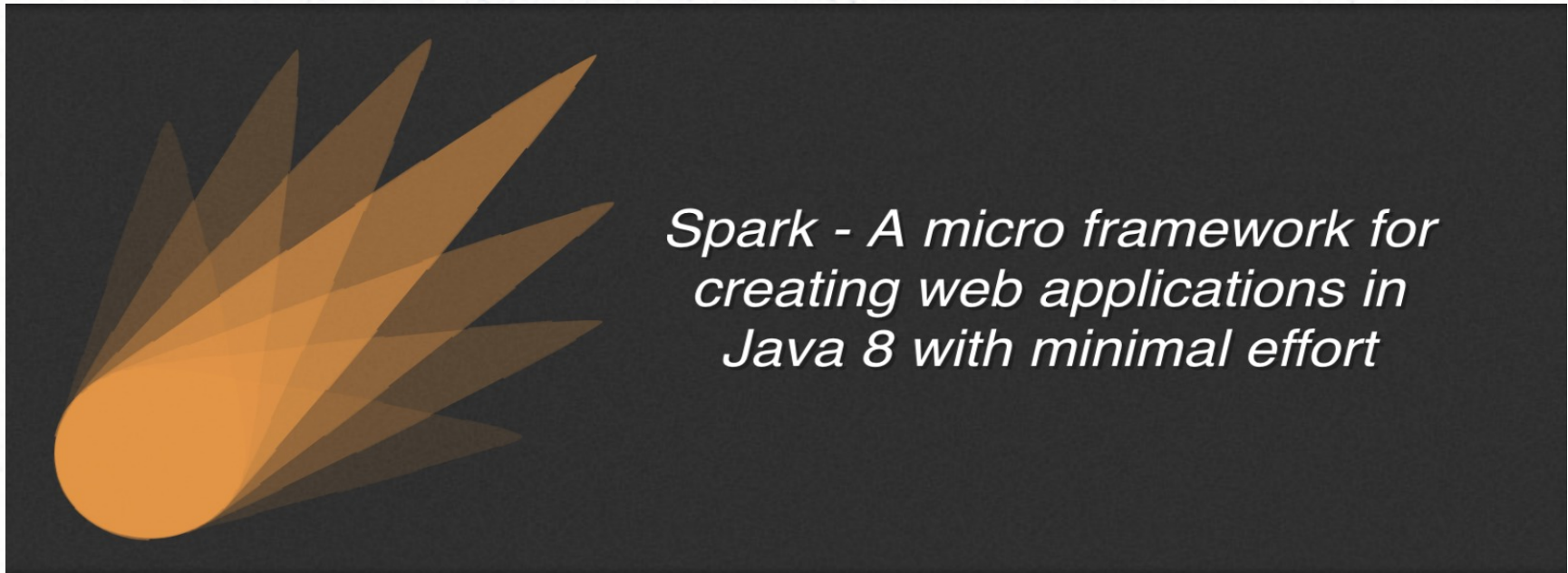
News headlines ?

Forecast ?



</>

WebApp



```

public class WebApp {

    /**
     * Sets up the routes and settings of the web application.
     *
     */
    public static void main(String[] args) {
        exception(Exception.class, (e, req, res) -> e.printStackTrace()); // print all exceptions

        // Configure Spark
        port(4567); // http://localhost:4567/ for local
        staticFileLocation("/public"); // HTML & CSS files

        // Routes

        // http://localhosts:4567/wiki?url=WIKIPEDIA_URL?articles=ARTICLES
        get("/wiki", (request, response) -> {
            String path = new File("src/main/java").getAbsolutePath();

            Parser parser = new Parser(path);
            String url = request.queryParams("url");
            String cloze = request.queryParams("cloze").toUpperCase();

            return parser.processSite(url, WordClass.valueOf(cloze));
        });

        // http://localhost:4567/stats
        get("/stats", (request, response) -> {
            return new ModelAndView(new HashMap(), "stats.mustache");
        }, new MustacheTemplateEngine());

        // http://localhost:4567/
        get("/", (request, response) -> {
            return new ModelAndView(new HashMap(), "index.html");
        }, new MustacheTemplateEngine());
    }
}

```


User Interface

```
1 <!DOCTYPE html>
2
3 <html>
4   <head>
5     <meta charset="utf-8" />
6     <title>Results</title>
7     <!-- <link type="text/css" rel="stylesheet" href="whatever.css" media="all" />  needs a stylesheet-->
8
9
10    <script src="http://code.jquery.com/jquery-1.10.2.min.js"></script>
11    <script src="json2.js"></script>
12
13
14    <script>
15
16      var userScoreArray = []; // create Global array
17
18      // fill sample data, just to get started
19      userScoreArray.push( new objScore('Fred', '2', '30') );
20      userScoreArray.push( new objScore('Frieda', '5', '75') );
21      console.log(userScoreArray);
22
23
24      function objScore(name, correct, total) {
25        this.name = name;
26        this.correct = correct;
27        this.total = total
```

Results

Enter your name:

Statistics

- Show immediate User feedback
- User can't change answer
- Final statistics page
- Changed languages
 - First: Java
 - Now: Javascript
 - Reason: efficiency

JUnit

- For Parser and Statistics
- At least one test for each method
- Simulated necessary conditions
- Special ParserTestVersion
 - Private → Public
- Problem: path-dependent tests