

# Contents

<b>Instructor Notes</b>	<b>1</b>
Timings Overview . . . . .	1
Rough Timings: . . . . .	1
<b>Exercises</b>	<b>2</b>
Exercise - Basics Overview Pre-requisites . . . . .	2
Exercise - Basics Overview . . . . .	2
Basic Concepts and Tooling Exercises . . . . .	2
Optional Exercises: Or Test REST Listicator in Buggy mode . . . . .	3
Exercises - Payload Object Abstractions . . . . .	3
Exercises - Abstractions . . . . .	3
RestMud . . . . .	4
Realease And Readiness Checklist . . . . .	4

## Instructor Notes

### Timings Overview

Planning:

09:00 - 17:00

- 09:00 - 10:30
- Break at 10:30 - 10:45 -10:45 - 12:00
- Lunch at 12:00 - 13:00
- 13:00 - 15:00
- Break at 15:00 - 15:15
- 15:15 - 17:00

‘use non speaking time for exercises and hands on’

### Rough Timings:

- 09:00 - 09:30 Workshop Overivew, Introductions, Start Basics
- 09:30 - 10:00 Basics Continue
- 10:00 - 10:30 REST Assured Automating slides, demo, exercises
- 10:30 - 10:45 - break
- 10:45 - 11:00 REST Assured Automating thru proxy, && exercises
- 11:00 - 11:30 Payload Objects and exercises
- 11:30 - 12:00 GSON & XStream, && exercises
- 12:00 - 13:00 - lunch
- 13:00 - 13:30 HTTP Calls with Java.net, comparison with REST Assured
- 13:30 - 14:00 Abstractions overview

- 14:00 - 14:30 Abstraction Exercises
- 14:30 - 15:00 Abstraction Exercises
- 15:00 - 15:15 - break
- 15:15 - 15:30 Overview of Bots
- 15:30 - 16:00 Writing a simple bot
- 16:00 - 16:30 Bot challenge on Rest Mud - write the best Scoring Bot
- 16:30 - 16:45 Performance Testing Discussion, Coverage Discussion
- 16:45 - 17:00 wrap up, feedback forms etc.

*Abstraction exercises probably take about 4 hours to complete fully*

## Exercises

---

### Exercise - Basics Overview Pre-requisites

- Download the source code, slides, handouts from
    - [compendiumdev.co.uk/page/atautores](http://compendiumdev.co.uk/page/atautores)
  - Download the app from
    - [github.com/eviltester/TestingApp/releases](https://github.com/eviltester/TestingApp/releases)
    - `compendium-of-test-apps-1-3-3.jar`
- 

### Exercise - Basics Overview

- Run the app
    - `java -jar compendium-of-test-apps-1-3-3.jar`
  - visit `localhost:4567/listicator/`
  - open the source code in IDE and run `RestAssuredBasicsTest` class
  - read over the REST Assured Docs
    - [github.com/rest-assured/rest-assured#documentation](https://github.com/rest-assured/rest-assured#documentation)
  - Experiment with some of the tests, try different JSONPath and XPath values
- 

### Basic Concepts and Tooling Exercises

- assuming the Testing App Running at `[http://localhost:4567/listicator/]`
- run the example tests `RestAssuredBasicsTest`
- get used to the API

- read the docs and experiment with Postman, Insomnia
  - GET, POST, DELETE, OPTIONS and POST
  - run the example tests through a proxy `RestAssuredProxyTest`
  - replay requests through a proxy
  - get used to the app and explore the HTTP Client functionality and explore the API based on its documentation
- 

## Optional Exercises: Or Test REST Listicator in Buggy mode

```
java -jar compendium-of-test-apps-1-3-3.r -bugfixes=false
```

- The system has been coded with some known bugs
- these are all fixed by default.
- start with `-bugfixes=false` to have known bugs
- See if you can find them

You can run the app twice on different ports to compare output, use the command line argument `-port` to start up the application on a different port e.g. `-port=1234` would start the app on port 1234

---

## Exercises - Payload Object Abstractions

- Using `RestAssuredMarshallingTest`
  - Run tests and make sure you understand them
  - It might help to run them through a proxy
  - see exercises at the bottom of the class file
- 

## Exercises - Abstractions

- create a `ServerConfig` which can be set by environment variables
    - `Map<String, String> env = System.getenv();`
  - allow configuration by System properties
    - `System.getProperties()`
    - `System.getProperty(String key, String def)`
    - allow System Properties to override environment variables
-

## RestMud

- <https://dashboard.heroku.com/apps/restmud>
  - Settings - change config vars for game secret code
  - <https://restmud.herokuapp.com/>
    - wiz / alanthewiz
    - change secretcode within wiz interface
      - \* <https://restmud.herokuapp.com/wiz>
- 

## Release And Readiness Checklist

- [ ] `pandocifier sh build-handouts.sh`
- [ ] create slides pdf using `pandocifier/export`
- [ ] `sh export-code.sh`
- [ ] update website from `release` folder
- [ ] workshop page is accessible
  - [compendiumdev.co.uk/page/atautorest](http://compendiumdev.co.uk/page/atautorest)
  - emergency download of main app available from page just in case
- [ ] can download the main test app
  - [github.com/eviltester/TestingApp](https://github.com/eviltester/TestingApp)
  - [github.com/eviltester/TestingApp/releases](https://github.com/eviltester/TestingApp/releases)
  - `compendium-of-test-apps-1-3-3.jar`
- [ ] can download the performance.zip
- [ ] can run the performance code from artillery
- [ ] can download the workshop code from the workshop page
- [ ] can unzip
- [ ] can load exercises and answers into editor
- [ ] can run exercises code against testing app locally
- [ ] can run answers code against testing app locally
- [ ] can download RestMud bot code from github and it works
  - [github.com/eviltester/restmudbots](https://github.com/eviltester/restmudbots)
- [ ] mailing list signup for advanced API automating exists
- [ ] mailing list form available on page