

Industrial strength Java/JEE Career Companion to open more doors

search here ...

Go

Home

Java FAQs

600+ Java Q&As

Career

Tutorials

Member

Why?

Can u Debug?

Java 8 ready?

Top X

Productivity Tools

Judging Experience?

Home › Interview › SQL, XML, UML, JSON, Regex Interview Q&A › YAML › YAML
with Java using the SnakeYaml library tutorial

YAML with Java using the SnakeYaml library tutorial

Posted on August 5, 2016 by Arulkumaran Kumaraswamipillai



Tweet



Share

Step 1: Add “snakeyaml” to maven pom.xml as a dependency

```

1
2 <dependency>
3   <groupId>org.yaml</groupId>
4   <artifactId>snakeyaml</artifactId>
5   <version>1.17</version>
6 </dependency>
7
```

Step 2: Define a YAML file under src/main/resources. For example, “logging.yml”

600+ Full Stack Java/JEE Interview Q&As ♥Free ♦FAQs

open all | close all

- ✚ Ice Breaker Interview
- ✚ Core Java Interview C
- ✚ JEE Interview Q&A (3
- ✚ Pressed for time? Jav
- ✚ SQL, XML, UML, JSC
- ✚ ERD (1)
- ✚ NoSQL (2)
- ✚ Regex (2)
- ✚ SQL (7)
- ✚ UML (1)
- ✚ JSON (2)
- ✚ XML (2)
- ✚ XSD (2)
- ✚ YAML (2)
- ✚ YAML with Java
- ✚ YAML with Sprin
- ✚ Hadoop & BigData Int
- ✚ Java Architecture Inte
- ✚ Scala Interview Q&As
- ✚ Spring, Hibernate, & I
- ✚ Testing & Profiling/Sa

```

1
2 paths:
3     logsPath: /mylogs
4
5 packagePrefix: com.myapp
6

```

[Other Interview Q&A 1](#)
[Free Java Interview](#)

Step 3: Create a LoggingConfiguration Java POJO class.

```

1
2 import java.util.Map;
3
4 public class LogConfiguration {
5
6     private Map<String, String> paths;
7     private String packagePrefix;
8
9     public Map<String, String> getPaths() {
10         return paths;
11     }
12
13     public void setPaths(Map<String, String> paths) {
14         this.paths = paths;
15     }
16
17     public String getPackagePrefix() {
18         return packagePrefix;
19     }
20
21     public void setPackagePrefix(String packagePrefix) {
22         this.packagePrefix = packagePrefix;
23     }
24
25     @Override
26     public String toString() {
27         return "LogConfiguration [paths=" + paths + ", packagePrefix=" + packagePrefix + "]";
28     }
29 }
30
31

```

Step 4: Finally, the main class that uses the Yaml library to read the configuration from the “logging.yml” file.

```

1
2 import java.io.InputStream;
3 import org.yaml.snakeyaml.Yaml;
4
5 public class TestYaml {
6
7     public static void main(String[] args) {
8
9         LogConfiguration config = null;
10
11         Yaml yaml = new Yaml();
12         try(InputStream in = ClassLoader.getSystemResourceAsStream("logging.yml")) {
13             config = yaml.loadAs(in, LogConfiguration.class);
14         } catch (Exception ex) {
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

16 Technical Key Areas

[open all](#) | [close all](#)

- [Best Practice \(6\)](#)
- [Coding \(26\)](#)
- [Concurrency \(6\)](#)
- [Design Concepts \(7\)](#)
- [Design Patterns \(11\)](#)
- [Exception Handling \(3\)](#)
- [Java Debugging \(21\)](#)
- [Judging Experience \(1\)](#)
- [Low Latency \(7\)](#)
- [Memory Management \(1\)](#)
- [Performance \(13\)](#)
- [QoS \(8\)](#)
- [Scalability \(4\)](#)
- [SDLC \(6\)](#)
- [Security \(13\)](#)
- [Transaction Management \(1\)](#)

80+ step by step Java Tutorials

[open all](#) | [close all](#)

- [Setting up Tutorial \(6\)](#)
- [Tutorial - Diagnosis \(2\)](#)
- [Akka Tutorial \(9\)](#)
- [Core Java Tutorials \(2\)](#)
- [Hadoop & Spark Tutorial \(1\)](#)
- [JEE Tutorials \(19\)](#)
- [Scala Tutorials \(1\)](#)
- [Spring & Hibernate Tutorial \(1\)](#)
- [Tools Tutorials \(19\)](#)
- [Other Tutorials \(45\)](#)

```

15         ex.printStackTrace();
16     }
17
18     String path = config.getPaths().get("logs");
19     System.out.println("path=" + path);
20
21     String packagePrefix = config.getPackagePrefix();
22     System.out.println("packagePrefix=" + packagePrefix);
23 }
24 }
25

```

Output:

```

1
2 path=/mylogs
3 packagePrefix=com.myapp
4

```

Learn YAML basics at [YAML with Spring & Java Q&As](#)

Popular Posts

◆ 11 Spring boot interview questions & answers

828 views

◆ Q11-Q23: Top 50+ Core on Java OOP Interview Questions & Answers

768 views

18 Java scenarios based interview Questions and Answers

400 views

001A: ◆ 7+ Java integration styles & patterns interview questions & answers

389 views

01b: ◆ 13 Spring basics Q8 – Q13 interview questions & answers

296 views

◆ 7 Java debugging interview questions & answers

293 views

01: ◆ 15 Ice breaker questions asked 90% of the time in Java job interviews with hints

286 views

◆ 10 ERD (Entity-Relationship Diagrams) Interview Questions and Answers

280 views

- [SQL Tutorial \(1\)](#)
- [Drools Tutorial \(5\)](#)
- [APache POI Tutorial \(1\)](#)
- [DOS Tutorial \(2\)](#)
- [Dozer Tutorial \(2\)](#)
- [ESB Tutorials \(1\)](#)
- [Groovy Beginner Tutorial \(1\)](#)
- [JasperReport Tutorial \(1\)](#)
- [Logging and auditing with Log4j \(1\)](#)
- [Metrics Tutorial \(2\)](#)
- [Orika Mapping Tutorial \(1\)](#)
- [Pentaho Kettle with Java \(1\)](#)
- [Unix Tutorial \(6\)](#)
- [Webspeher MQ Tutorial \(1\)](#)
- [YAML Tutorial \(1\)](#)
- [YAML with Java](#)

100+ Java pre-interview coding tests

[open all](#) | [close all](#)

- [Can you write code? \(1\)](#)
- [◆ Complete the given code \(1\)](#)
- [Converting from A to B \(1\)](#)
- [Designing your class \(1\)](#)
- [Java Data Structures \(1\)](#)
- [Passing the unit tests \(1\)](#)
- [What is wrong with this code? \(1\)](#)
- [Writing Code Home Assignment \(1\)](#)
- [Written Test Core Java \(1\)](#)
- [Written Test JEE \(1\)](#)

How good are your?

[open all](#) | [close all](#)

- [Career Making Knowledge \(1\)](#)
- [Job Hunting & Resumes \(1\)](#)

♦ Q24-Q36: Top 50+ Core on Java classes, interfaces and generics interview questions & answers

240 views

001B: ♦ Java architecture & design concepts interview questions & answers

202 views

Bio

Latest Posts



Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via [Amazon.com](https://www.amazon.com) in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site. **945+** paid members. [join my LinkedIn Group](#). [Reviews](#)



About Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via [Amazon.com](https://www.amazon.com) in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site. **945+** paid members. [join my LinkedIn Group](#). [Reviews](#)

◀ 03: Q13 – Q18 Scala FP combinators interview questions & answers

05: Q37-Q41 – Data lake & metadata interview questions & answers ▶

Posted in YAML, YAML Tutorial

Empowers you to open more doors, and fast-track

Technical Know Hows

☀ [Java generics in no time](#) ☀ [Top 6 tips to transforming your thinking from OOP to FP](#) ☀ [How does a HashMap internally work? What is a hashing function?](#)

☀ [10+ Java String class interview Q&As](#) ☀ [Java auto un/boxing benefits & caveats](#) ☀ [Top 11 slacknesses that can come back and bite you as an experienced Java developer or architect](#)

Non-Technical Know Hows

☀ [6 Aspects that can motivate you to fast-track your career & go places](#) ☀ [Are you reinventing yourself as a Java developer?](#) ☀ [8 tips to safeguard your Java career against offshoring](#) ☀ [My top 5 career mistakes](#)

Prepare to succeed

☀ [Turn readers of your Java CV go from “Blah blah” to “Wow”?](#) ☀ [How to prepare for Java job interviews?](#) ☀ [16 Technical Key Areas](#) ☀ [How to choose from multiple Java job offers?](#)

Select Category ▼

© Disclaimer

The contents in this Java-Success are copy righted. The author has the right to correct or enhance the current content without any prior notice.

These are general advice only, and one needs to take his/her own circumstances into consideration. The author will not be held liable for any damages caused or alleged to be caused either directly or indirectly by these materials and resources. Any trademarked names or labels used in this blog remain the property of their respective trademark owners. No guarantees are made regarding the accuracy or usefulness of content, though I do make an effort to be accurate. Links to external sites do not imply endorsement of the linked-to sites.