Java Programming I

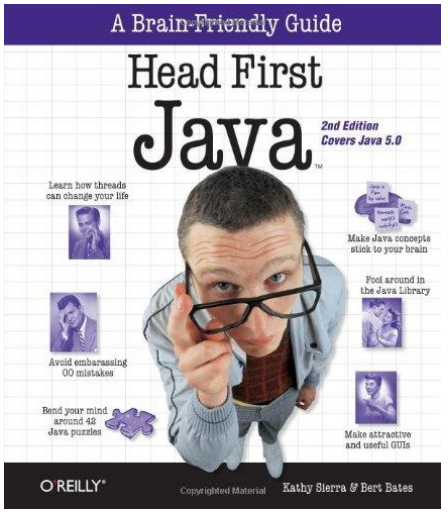
Introduction

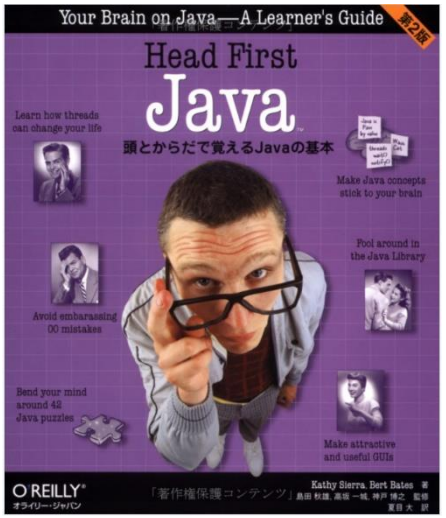
Administration

- Course home page:
  http://web-int.u-aizu.ac.jp/~vkluev/courses/javaone/
- Instructors:
  - Prof. Vitaly Klyuev
    - E-mail: vkluev@u-aizu.ac.jp
  - Associate Prof. Yuichi Yaguchi
    - E-mail: yaguchi@u-aizu.ac.jp
  - Senior Associate Prof. Mohamed Hamada
    - E-mail: hamada@u-aizu.ac.jp
- Class materials and assignments will be put on the Web every Friday
- Please print them and take to our class
- Basic for grades:
  - Assignments 50%,
  - Quizzes during lectures 15%,
  - Final examination 35%.

Literature

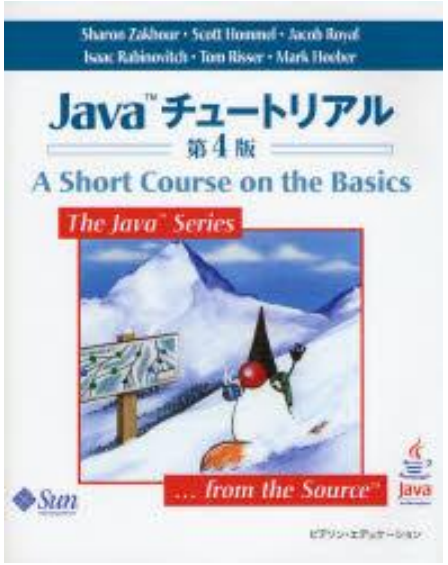
- Head First Java, 2nd Edition, by Kathy Sierra and Bert Bates, O'REILLY, 2005.
- Head First Java 第2版 —頭とからだで覚えるJavaの基本 大型本 by Kathy Sierra, et al., 2006.

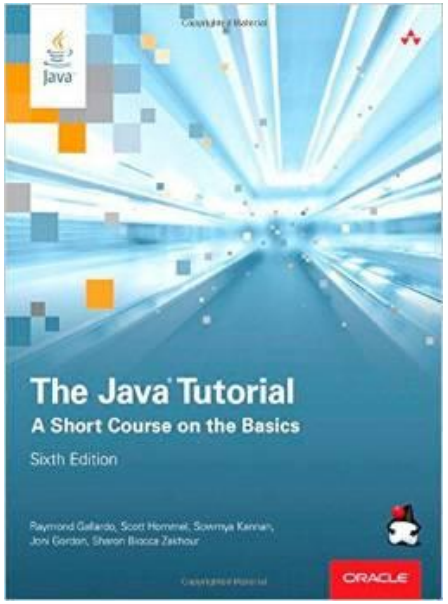




Literature

- Javaチュートリアル 第4版 (The Java Series) 単行本 – November, 2007.
- The Java Tutorial: A Short Course on the Basics (6th Edition) (Java Series) by R. Gallardo, et al., 2014.
  - Electronic version: http://docs.oracle.com/javase/tutorial/



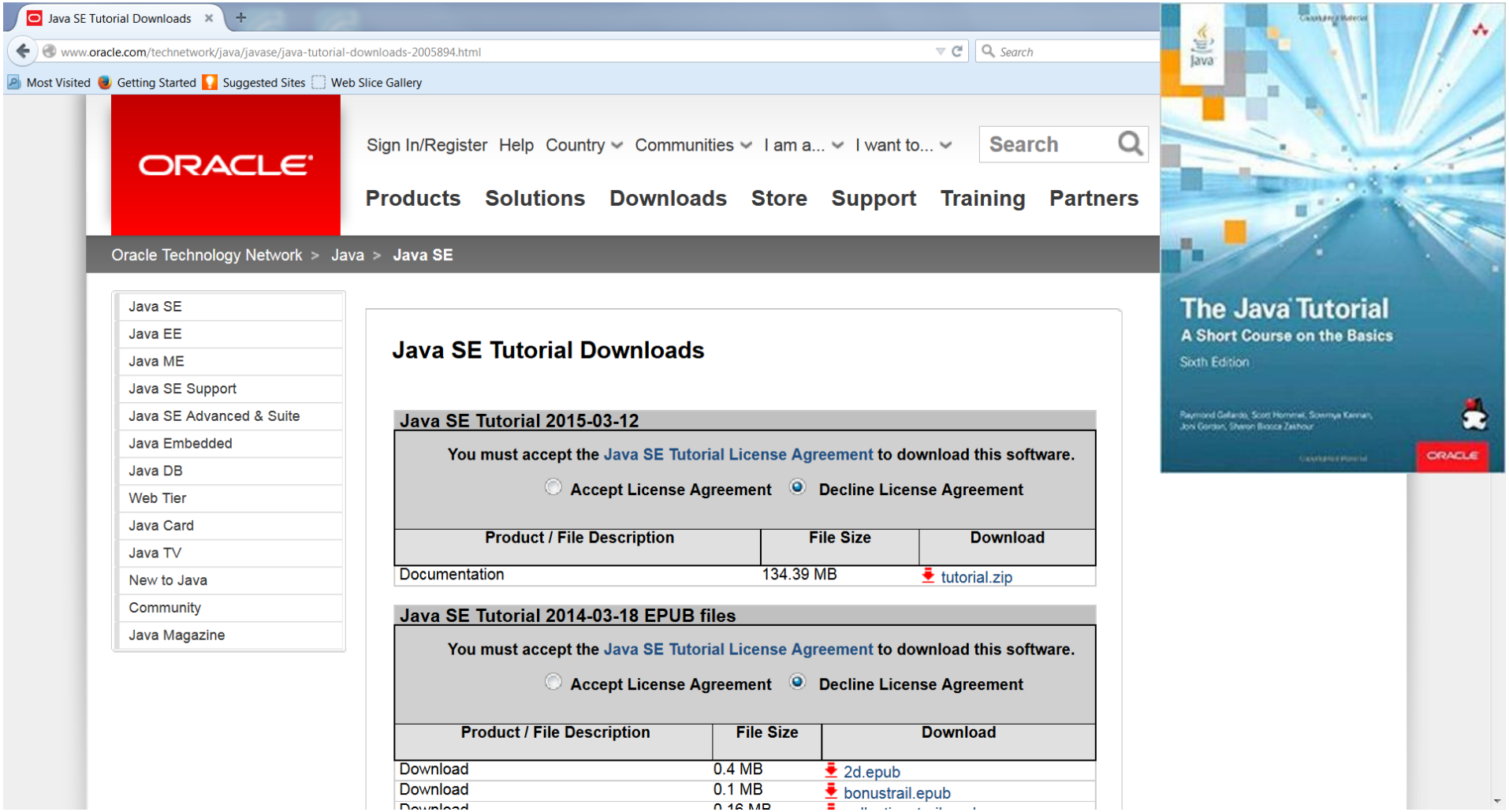


Online Documentation

- Java provides online documentation for the whole environment:
  - How to compile and execute programs;
  - JDK (Java Development Kit) classes and their methods;
  - Many example programs;
  - Many documents that address different topics in Java.

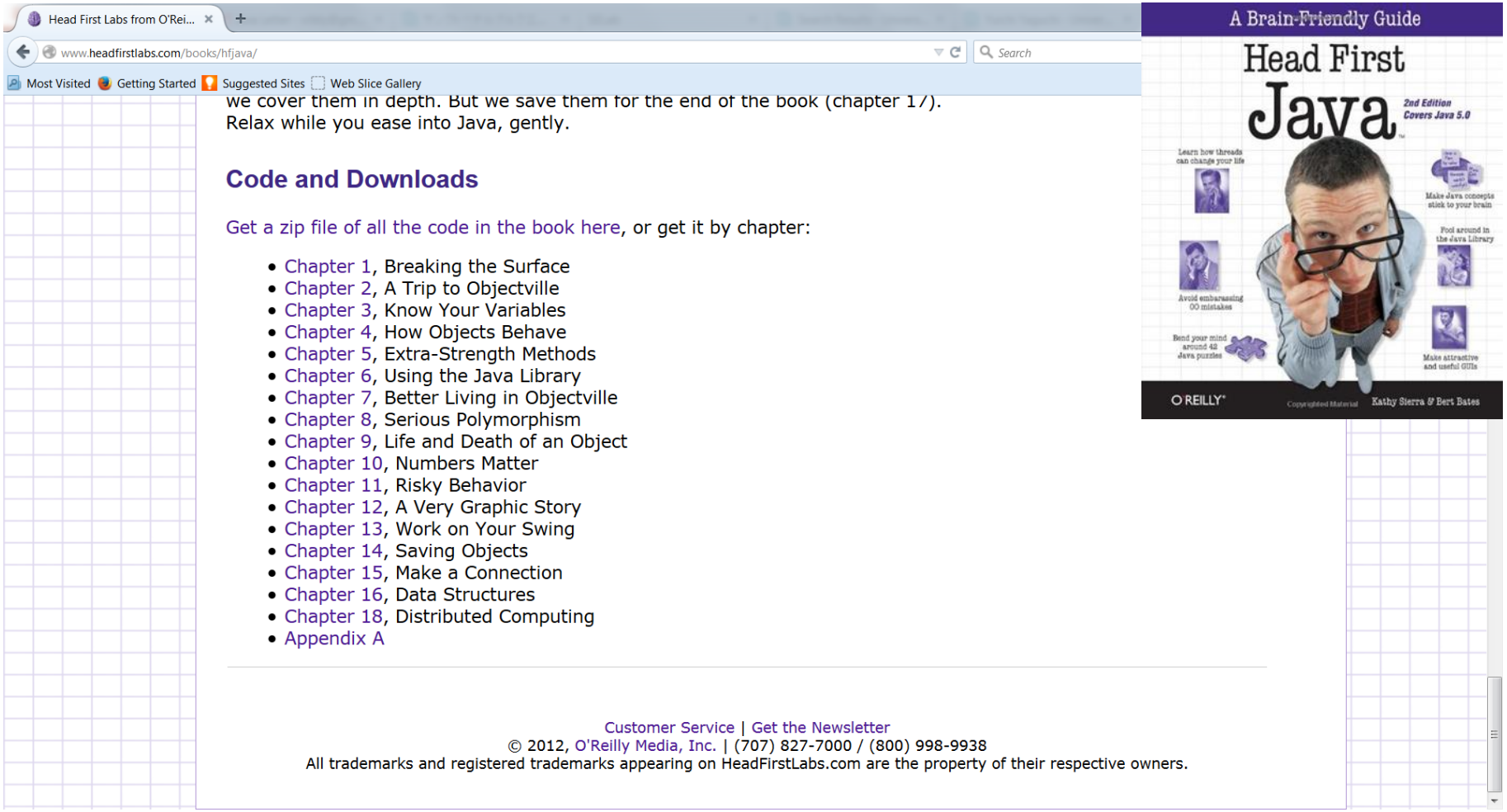
Java Tutorials

http://www.oracle.com/technetwork/java/javase/java-tutorial-downloads-2005894.html



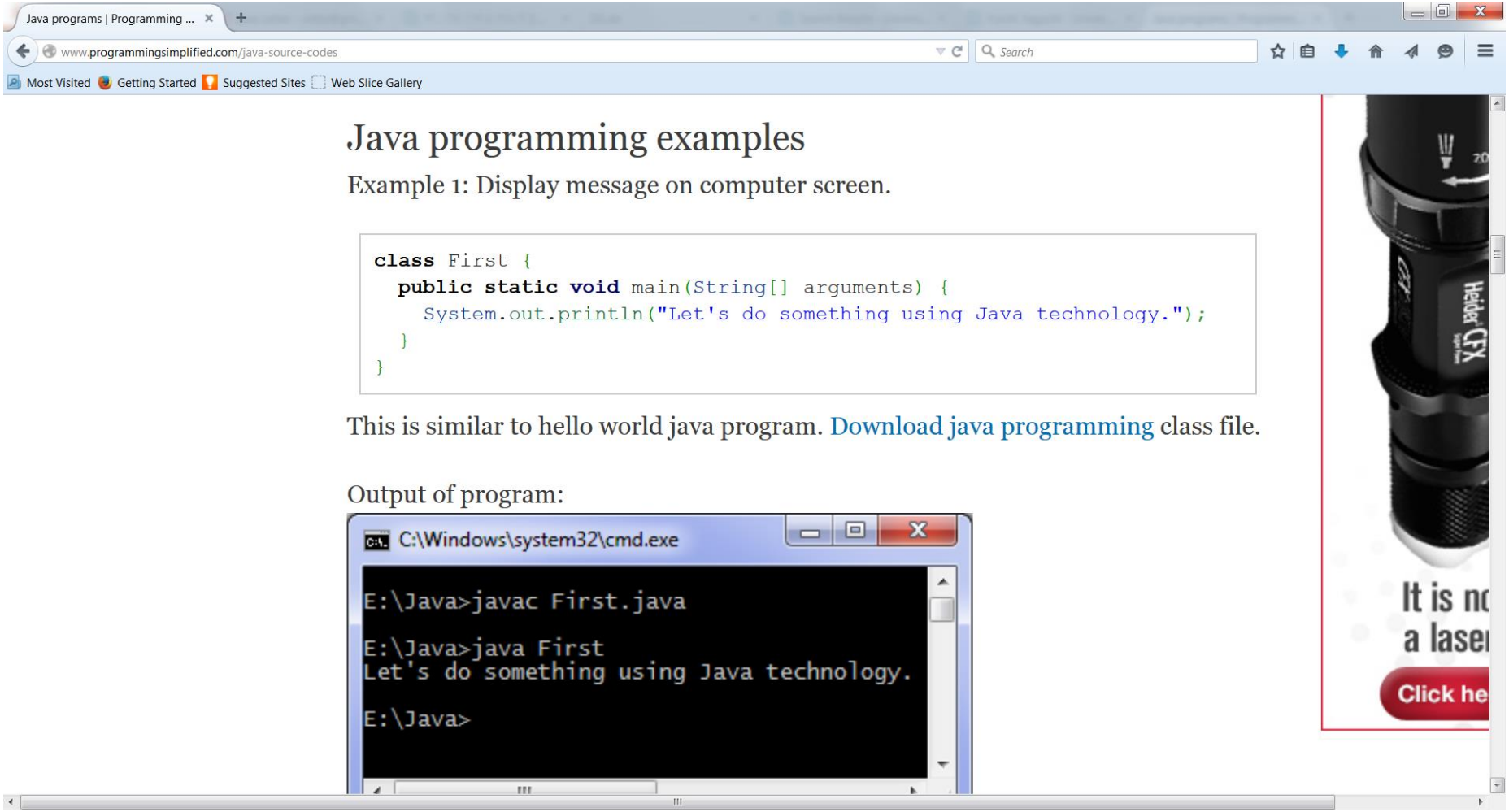
Java Code Examples

http://www.headfirstlabs.com/books/hfjava/



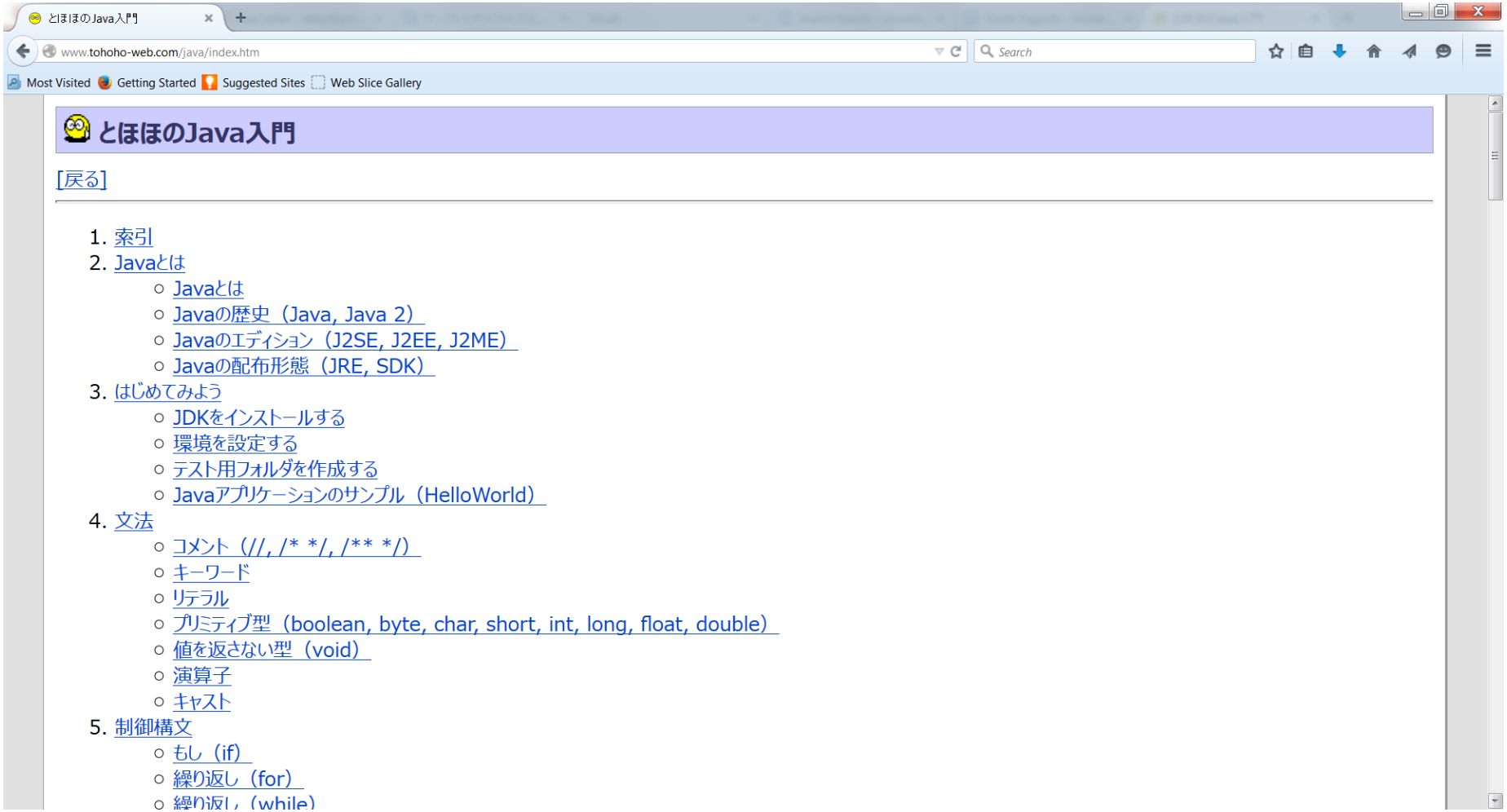
Java Code Examples

http://www.programmingsimplified.com/java-source-codes



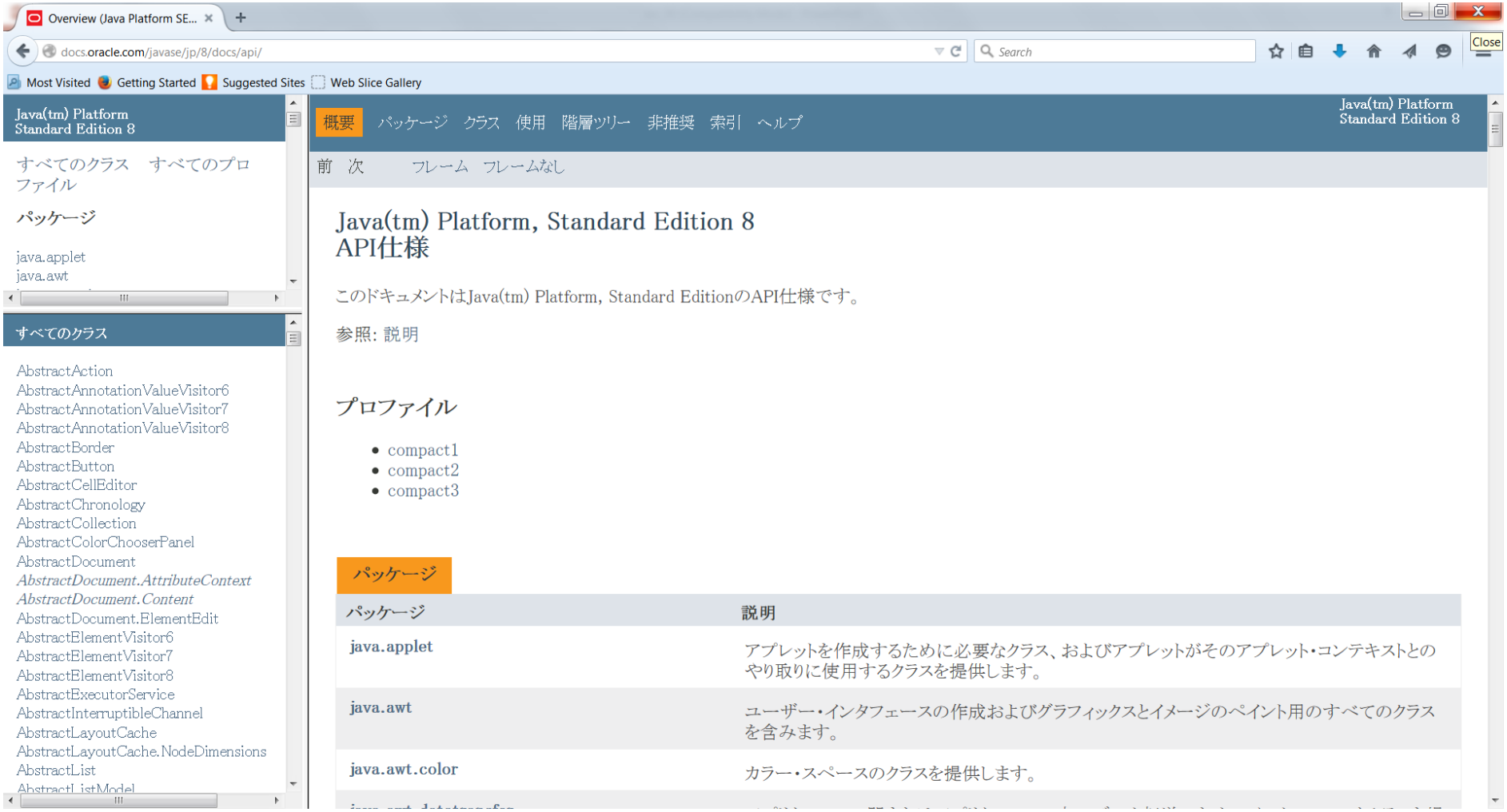
Java on-line resource in Japanese

http://www.tohoho-web.com/java/index.htm



Java API
(Application Programming Interface)

http://docs.oracle.com/javase/jp/8/docs/api/



Jeliot

Jeliot is a system for animating programs in Java. It will help you understand how your code is executed.

http://cs.joensuu.fi/jeliot/index.php

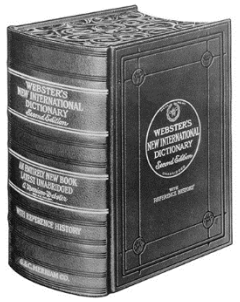
How to study course materials

1. Read lecture slides (understand ALL examples);
2. Run examples from the lectures using *Jeliot*

   http://web-int.u-aizu.ac.jp/~yaguchi/courses/Java1/2016/sp_care_01.html
3. Read the corresponding chapters in the text book;
4. Study a Special Care Sets on the course Webpage;
5. Work on exercises.

How to work on exercises

- Read problem descriptions very carefully: You MUST read ALL sentences.
- To understand content, you may use
  - paper-based dictionaries;
  - electronic dictionaries;
  - Web-based dictionaries
    - –http://www.alc.co.jp/



How to work on exercises

- NEVER use translation systems such as google translate!
  - They NEVER will translate the details of the description;
  - In software development, details are very much important!
  - A Japanese description, obtained as the result of translation by a translation systems, specifies a different problem, not the problem you should solve!

How to work on exercises

- Go back to the lecture examples to find the analogies between them and the problem you should solve:
  - Make the bridges between them in your brain.
- When code is prepared, analyze the results produced by your code to make sure that they are correct.
- Following these instructions, you become INDEPENDENT!

Requirements for the Lectures

- Please print our the slides BEFORE every lecture:
  - http://web-int.u-aizu.ac.jp/~vkluev/courses/javaone
- There will be a quiz at EVERY lecture (to answer the questions, you have to have the lecture slides)
- Quiz sheets will be available ONLY at the lecture time. It is not allowed to get the quiz questions after the lecture.
- Quiz sheets will be collected at the end of the class: One student can submit only one sheet.

How to get credits for the exercises

- To get credits for your solution, you should
  - show your TA or your instructor how your programs work;
  - answer their questions;
  - submit your source code to your instructor.

Exercise grading policy

- The penalty for the late submissions is 30% of points for each problem;
- Late submissions are acceptable within two weeks after the deadline;
- Three weeks or later after the deadline, the score for the late submission is zero.