**Assignment1**

**Part 1**

**Distributed Version Control with Git and Bitbucket, and Development Environments**

***Carefully*** go through the following steps: http://www.qcitr.com/usefullinks.htm#lesson3b

**Part 2**

**Development Installation:**

**Note: *Print*** the following tutorials, and ***keep*** them for the remainder of the course!

1. **AMPPS:** (http://www.ampps.com/)
   a. Tour: http://www.ampps.com/tour
   b. Download and Installation: http://www.ampps.com/wiki/Main_Page
   c. Installation Demo: http://www.ampps.com/demo

   **NOTE: Mac Users with Monterey or higher, See Mac Installation Instructions!**

2. Download and install **JDK:**
   https://personal.ntu.edu.sg/ehchua/programming/howto/JDK_HowTo.html
   **Windows: Stop *after*** "Step 6: Compile and Run the Hello-World Java Program"
   **Mac: Stop *after*** "Step 4: Compile and Run the Hello-World Java Program"

3. Download and install Apache **Tomcat:**
   https://personal.ntu.edu.sg/ehchua/programming/howto/Tomcat_HowTo.html
   **Stop after completing**:"2.6 STEP 5: Develop and Deploy a WebApp"

4. Using Git Bash (or Terminal), cd to local **tomcat/webapps** subdirectory:
   Example (**NOTE:** your path may be different): **cd C:/tomcat/webapps**

5. Clone assignment starter files:
   **git clone https://bitbucket.org/mjowett/lis4368_student_files lis4368**
   (clones **lis4368_student_files** Bitbucket repo to local lis4368 subdirectory)

**Helper video (Student Starter Files):** http://qcitr.com/vids/lis4368_starter_files.mp4

6. Review subdirectories and files

7. ***Must* delete .git subdirectory (*be sure* to show hidden files/folders)**
   (If you do not delete the .git subdirectory you will get an error when attempting to push files!)

8. Open **index.jsp** and review code:
   a. Suitably modify meta tags
   b. Change title, navigation links, and header tags appropriately
   c. Include JDK **and** Tomcat installation screenshots in **a1/img subdirectory**.
   d. Research and create your *own* **favicon** (e.g., use your initials).

9. Push your local repository to the one hosted by Bitbucket's servers: See **Part 1** (above).

10. Provide me with **read-only** access to Bitbucket repository: See **Part 1** (above).

**Note:**

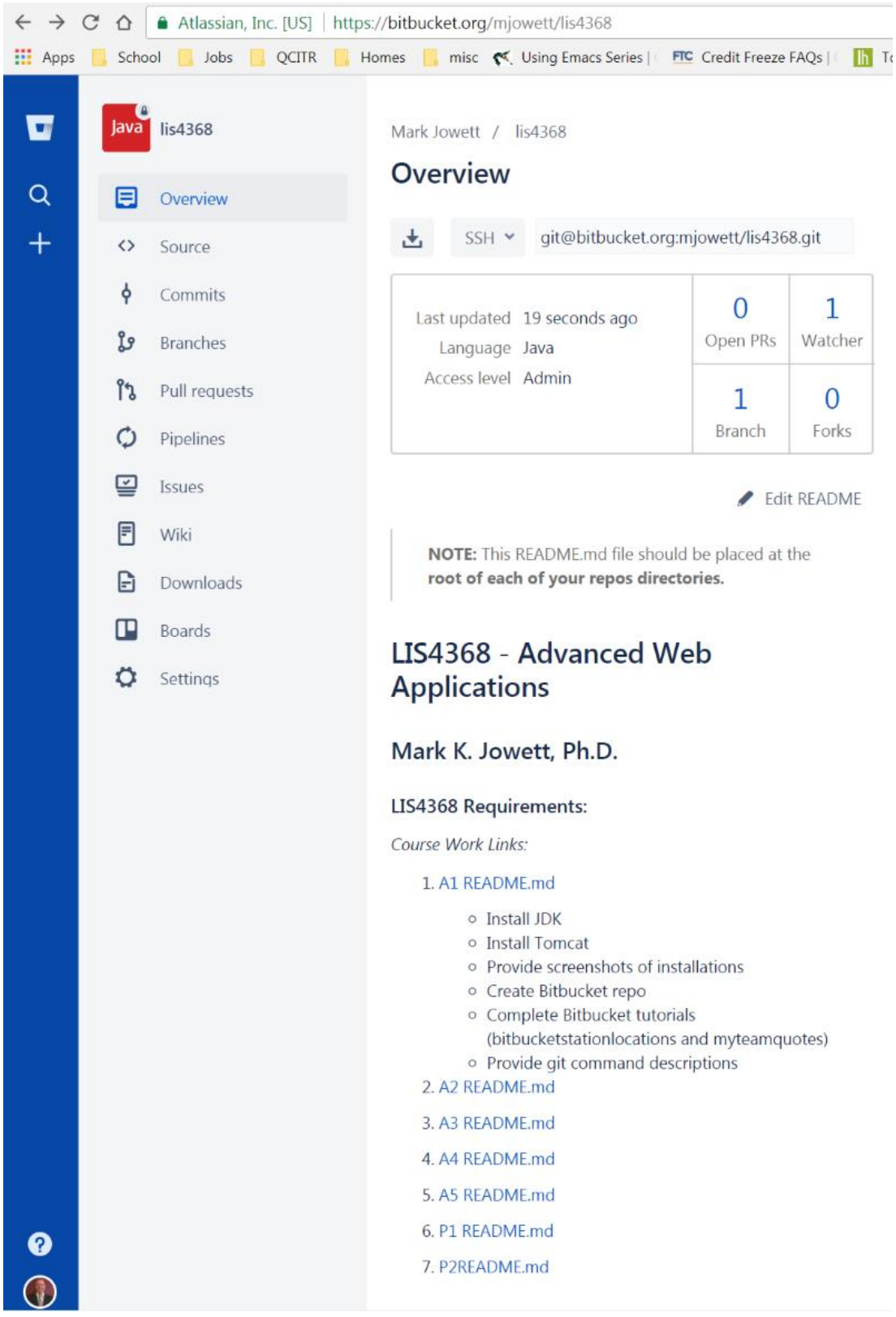
**README.md** file should include the following items (**see screenshots below**):

1. Screenshot of running **java Hello** (#1 above);
2. Screenshot of running http://localhost:9999 (#2 above, Step #4(b) in tutorial);
3. Screenshot of a1/index.jsp
4. git commands w/short descriptions;
5. Bitbucket repo links:
   a. This assignment, and
   b. The completed tutorial repo above (**bitbucketstationlocations**).
   (See link in screenshot below.)

**Deliverables (see screenshots below):**

1. Provide **Bitbucket** read-only access to **lis4368** repo, include links to the repo (**BitbucketStationLocations**) you created in the above tutorials in **README.md**, using Markdown syntax, (**README.md** must also include screenshots per above.)
   (**DO NOT create README OR .gitignore in Bitbucket**—ALWAYS do so locally, then push them to Bitbucket.)
2. **FSU's Learning Management System**: include lis3781 **Bitbucket** repo link

**Main README.md**



**A1 README.md**
**(Note: older Tomcat version – display newer version)**

Atlassian, Inc. [US] | https://bitbucket.org/mjowett/lis4368/src/87542a70ff134

Apps | School | Jobs | QCITR | Homes | misc | Using Emacs Series | FTC Credit Fr

Mark Jowett / lis4368 / Source

lis4368

Overview
Source
Commits
Branches
Pull requests
Pipelines
Issues
Wiki
Downloads
Boards
Settings

## README.md

master | lis4368 / a1 / README.md

8754287 3 minutes ago | Full commit

**NOTE:** This README.md file should be placed at the **root of each of your repos directories.**

Also, this file **must** use Markdown syntax, and provide project documentation as per below--otherwise, points **will** be deducted.

### LIS4368 - Advanced Web Applications Development

Mark K. Jowett, Ph.D.

**Assignment 1 Requirements:**

*Three Parts:*

1. Distributed Version Control with Git and Bitbucket
2. Java/JSP/Servlet Development Installation
3. Chapter Questions (Chs 1 - 4)

**README.md file should include the following items:**

- Screenshot of running java Hello (#1 above);
- Screenshot of running http://localhost:9999 (#2 above, Step #4(b) in tutorial);
- git commands w/short descriptions;
- Bitbucket repo links: a) this assignment and b) the completed tutorial above (bitbucketstationlocations).

> This is a blockquote.
>
> This is the second paragraph in the blockquote.
>
> Git commands w/short descriptions:

1. git init - definition goes here...
2. git status
3. git add
4. git commit
5. git push
6. git pull
7. One additional git command

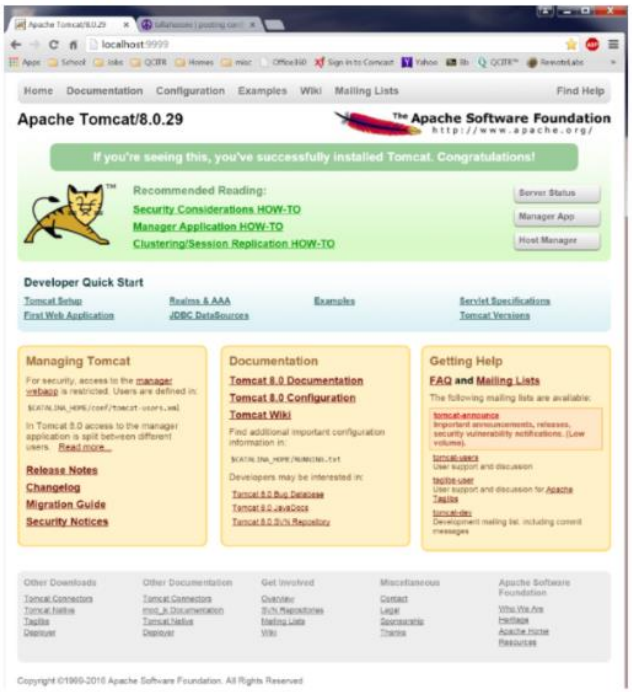
**Assignment Screenshots:**

*Screenshot of running java Hello:*

```
E:\Courses\_LIS4368>javac Hello.java

E:\Courses\_LIS4368>java Hello
Hello, world!
```

*Screenshot of running http://localhost:9999:*

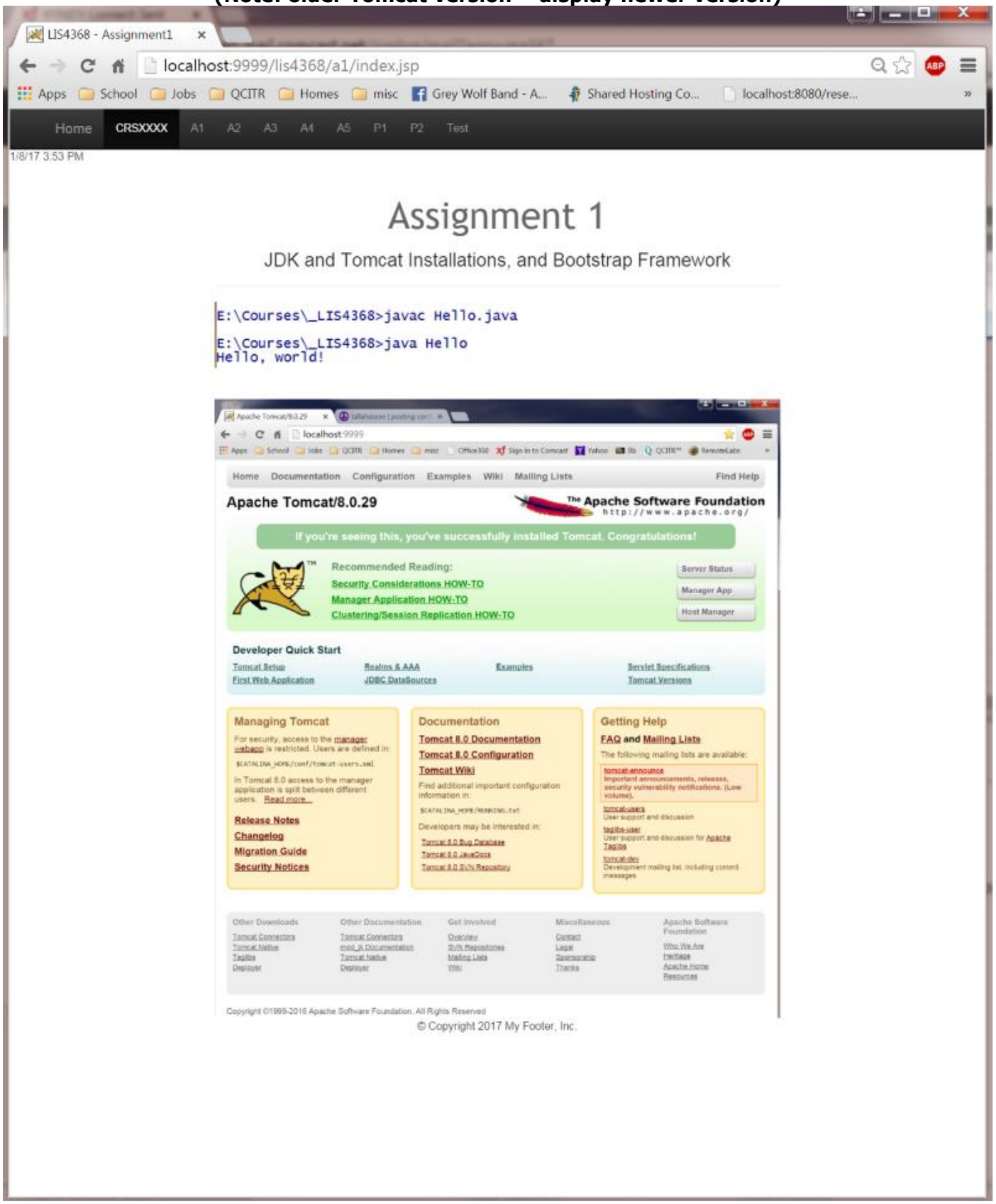



Tutorial Links:

*Bitbucket Tutorial - Station Locations:* A1 Bitbucket Station Locations Tutorial Link

*Tutorial: Request to update a teammate's repository:* A1 My Team Quotes Tutorial Link

**A1 index.jsp**
**(Note: older Tomcat version – display newer version)**



**Part 3**

**Answer the following questions (Chs. 1 - 4):**

**Note: Do *not* install any additional software indicated in the textbook.**

1. A JavaBean, or bean, is a Java class that

a.declares public instance variables
b.doesn't provide a zero-argument constructor
c.doesn't provide get and set methods for all of it's private instance variables that follow standard Java naming conventions
d.none of the above

2. A servlet

a.runs in a JSP
b.runs on a server
c.runs on the client
d.runs within HTML code

3. A set of pages that are generated in response to user requests is called a/an

a.servlet
b.JSP
c.domain name
d.web application

4. A web application is a type of

a.client/server application
b.database application
c.JSP application
d.Java application

5. After you edit a web.xml file, you can check to see if it is still valid by checking it against its

a.Output window
b.Projects window
c.XML schema
d.deployment descriptor

6. An HTML form contains one or more controls like

a.text boxes and check boxes
b.radio buttons
c.combo boxes and list boxes
d.all of the above

7. An HTTP request

a.is sent from the web browser to a web server
b.is sent from the web browser to the client
c.is sent from the servlet engine to the client
d.is sent from the client to the servlet engine

8. For web applications, data can be stored in

a.text files
b.databases
c.XML files
d.all of the above

9. In HTML, these do not display on their own lines in browsers.

a.block elements
b.inline elements
c.both a and b
d.none of the above

10. Since the web.xml file describes how the web application should be configured when it is deployed on a server, the file is known as the

a.the request object
b.style sheet (SS)
c.app config file (ACD)
d.deployment descriptor (DD)

11. The HTML5 semantic elements include the

a.header and section elements
b.nav elements
c.aside and footer elements
d.all of the above

12. The controller manages the flow of the application, and this work is done by one or more

a.request objects
b.servlets
c.JSPs
d.data access classes

13. The presentation layer for a typical servlet/JSP web application consists of

a.Java classes
b.servlets
c.HTML pages and JSPs
d.data access classes

14. The specification that describes how web servers can interact with all Java web technolgies is know as

a.The  Java Development Kit (JDK)
b.The Java Enterprise Edition (Java EE)
c.The Application Server
d.The HyperText Transfer Protocal

15. To develop Java web applications, you can use an Integrated Development Environment (IDE) such as:

a.NetBeans or Eclipse
b.Foxfire or Netscape
c.FileZilla or CuteFTP
d.Tomcat or Apache

16. To transfer your web application files to a web server, you can use a/an _____ client such as FileZilla.

a.HTTP (Hypertext Transfer Protocol)
b.JDK (Java Development Kit)
c.HTML (Hypertext Markup Language)
d.FTP (File Transfer Protocol)

17. When you use the MVC pattern, you

a.make each layer as independent as possible
b.perform all data validation on the client
c.use the pattern for every page in the application
d.all of the above

18. Which of the following is not a benefit of using the MVC pattern for an application?

a.it’s easier to test and debug the application
b.JSPs are used for what they do best
c.servlets are used for what they do best
d.it requires less code

19. Which statement is true for the order in which styles override other styles when you use CSS?

a. more specific styles override less specific styles
b. less specific styles override more specific styles
c. the first style that's applied overrides following styles

20. You can use tables to organize data in

a. rows and columns
b. drawers
c. inline and block elements
d. none of the above

21. To minimize the amount of Java code in your JSPs, you

a. use JSPs to perform the business logic
b. use JSPs to define the user interface
c. use servlets to handle the processing requirements
d. use servlets to return HTML to the browser

22. When you use the MVC pattern, the controller directs the flow of control to

a. the browser and the model
b. the view and the model
c. the view, model, and data store
d. the model and data store

23. Which directory stores the Java classes of a servlet/JSP application?

a. WEB-INF
b. WEB-INF\classes
c. META-INF
d. META-INF\classes

24. Which of the following is an API for working with databases?

a. JPA
b. JSF
c. EJB
d. Spring