“Современные технологии программирования”

Лабораторная работа #02

Цель работы: Изучить процесс установки инструментального ПО и настройка окружения для разработки веб-приложений на язык Java (JDK, TomCat, Maven). Реализация и запуск на сервере приложений Java простого сервлета (Servlet) передаюющего на клиент (в браузер) статический HTML-контент (сайт полученный в результате выполнения лабораторной 1). Рассмотреть два варианта конфигурации сервлета с использованием XML файла, а также на основе аннотации сигнала.

* 1. Подготовительная часть:
  2. Скачать и установить актуальную версию (1.8.Х или выше) инструментальных средств разработки для языка Java: Java Development Kit.

**Java Development Kit (JDK)** — бесплатно распространяемый компанией Oracle Corporation (ранее Sun Microsystems) комплект разработчика приложений на языке Java, включающий в себя компилятор Java (javac), стандартные библиотеки классов Java, примеры, документацию, различные утилиты и исполнительную систему Java (JRE). В состав JDK не входит интегрированная среда разработки на Java, поэтому разработчик, использующий только JDK, вынужден использовать внешний текстовый редактор и компилировать свои программы, используя утилиты командной строки.

– Скачать в учебной аудитории можно по ссылке:

[\\5-704-SERVER2\Soft\\_поПредметам\СТП\Java\JDK\](file:///\\5-704-SERVER2\Soft\_поПредметам\СТП\Java\JDK\)

дома с официального сайта:

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

или по запросу «jdk 8 download» в поисковой системе.

*Внимание! Для компиляции программ и работы с утилитами JDK из командной строки после непосредственной установки инструментов необходимо будет ОБЯЗАТЕЛЬНО ПРОПИСАТЬ СООТВЕТСТВУЮЩИЕ ПЕРЕМЕННЫЕ СРЕДЫ ОКРУЖЕНИЯ ОС! Данный этап будет детально рассмотрен далее!*

*Примечание! В примере используется версия «jdk-8u73-windows-x64», а также рассматривается путь установки по умолчанию ”C:\Program Files\Java\jdk1.8.0\_73” (для 64-битной версии) и C:\Program Files(x86)\Java\jdk1.8.0\_73 (для 32-битной версии). Если у Вас будет изменен путь установки или версия, то следующие пункты касающие переменных среды окружения необходимо выполнять с учетом данных изменений!*

* 1. Cкачать и установить на диск актуальную версию контейнера сервлетов Apache Tomcat (8.5 или выше).

**Apache Tomcat** (в старых версиях — Catalina) — контейнер сервлетов с открытым исходным кодом, разрабатываемый Apache Software Foundation. Реализует спецификацию сервлетов и спецификацию JavaServer Pages (JSP) и JavaServer Faces (JSF). Написан на языке Java. **Tomcat позволяет запускать веб-приложения**, содержит ряд программ для самоконфигурирования.

– Скачать в учебной аудитории можно по ссылке:

[\\5-704-SERVER2\Soft\\_поПредметам\СТП\Java\Servers\](file:///\\5-704-SERVER2\Soft\_поПредметам\СТП\Java\Servers\)

дома с официального сайта:

<https://tomcat.apache.org/download-90.cgi>

или по запросу «apache tomcat 9 download» в поисковой системе.

Примечание! В примере используется версия «apache-tomcat-9.0.0.M26-windows-x64»

– Процесс установки Tomcat заключается в распаковке содержимо архива в произвольную директорию на диске. Рекомендуется распаковать содержимое по соседству с JDK. Например, в папку на диске “C:\Program Files\Java\ apache-tomcat-9.0.0.M26” (Рис. 1).

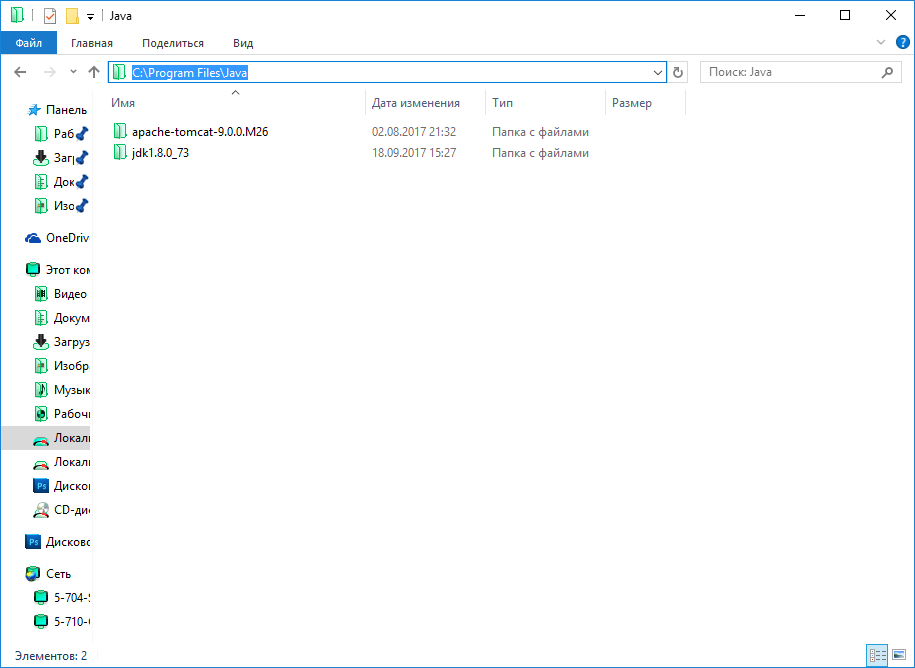


Рис. 1

*Внимание! Для работы с Tomcat из командной также необходимо будет прописать соответствующие переменные среды окружения ос! Данный этап будет детально рассмотрен далее!*

* 1. Cкачать и установить на диск актуальную версию системы автоматизированной сборки приложений Apache Maven (3.5 или выше).

**Apache Maven** (Maven)– это инструмент для сборки Java проекта: компиляции, создания jar, создания дистрибутива программы, генерации документации. Простые проекты можно собрать в командной строке. Если собирать большие проекты с командной строки, то команда для сборки будет очень длинной, поэтому её иногда записывают в bat/sh скрипт. Но такие скрипты зависят от платформы (тип процессора + тип ОС). Для того чтобы избавиться от этой зависимости и упростить написание скрипта используют инструменты для сборки проекта. Для платформы Java существуют два основных инструмента для сборки: Ant и Maven.

Основные преимущества Maven:

* Независимость от OS. Сборка проекта происходит в любой операционной системе. Файл проекта один и тот же.
* Управление зависимостями. Редко какие проекты пишутся без использования сторонних библиотек (зависимостей). Эти сторонние библиотеки зачастую тоже в свою очередь используют библиотеки разных версий. Мавен позволяет управлять такими сложными зависимостями. Что позволяет разрешать конфликты версий и в случае необходимости легко переходить на новые версии библиотек.
* Возможна сборка из командной строки. Такое часто необходимо для автоматической сборки проекта на сервере (Continuous Integration).
* Хорошая интеграция со средами разработки. Основные среды разработки на java легко открывают проекты которые собираются c помощью maven. При этом зачастую проект настраивать не нужно - он сразу готов к дальнейшей разработке.
* Как следствие если с проектом работают в разных средах разработки, то maven удобный способ хранения настроек. Настроечный файл среды разработки и для сборки один и тот же - меньше дублирования данных и соответственно ошибок.
* Декларативное описание проекта.

– Скачать в учебной аудитории можно по ссылке:

[\\5-704-SERVER2\Soft\\_поПредметам\СТП\Java\ BuildTool\](\\\\5-704-SERVER2\\Soft\\_поПредметам\\СТП\\Java\\BuildTools\\)

дома с официального сайта:

[http://maven.apache.org/download.cgi](https://tomcat.apache.org/download-90.cgi)

или по запросу «apache maven 3.5 download» в поисковой системе.

*Примечание! В примере используется версия «apache-maven-3.5.0-bin.zip»*

– Процесс установки Maven заключается в распаковке содержимо архива в произвольную директорию на диске. Рекомендуется распаковать содержимое по соседству с JDK. Например, в папку на диске “C:\Program Files\Java\apache-maven-3.5.0“ (Рис. 2)

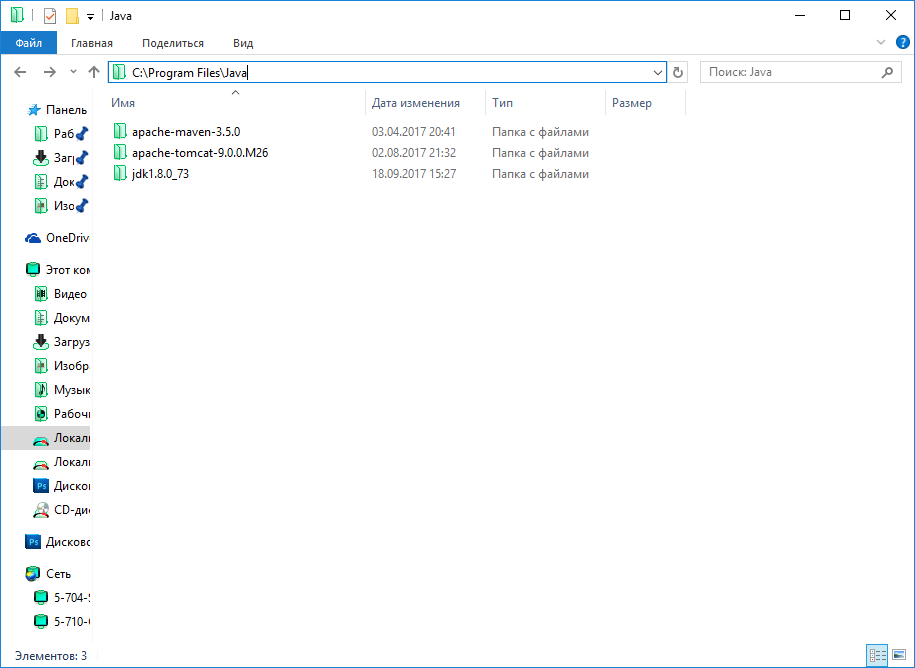
”.

Рис. 2

*Внимание! Для работы с Maven из командной также необходимо будет прописать соответствующие переменные среды окружения ос! Данный этап будет детально рассмотрен далее!*

* 1. Добавить переменные среде окружения для всех перечисленных в пункте 1-4 средств разработки и убедиться в их работоспособности из интерпретатора командной строки (консоли) «cmd».

Переменная окружения (переменная среды, англ. environment variable) в Windows используются для хранения текстовых строк пользователя и информации о настройках операционнойой системы. Одной из наиболее важных перемнных среды окружения является переменная PATH, в которой указываются пути поиска исполняемых файлов. Поэтому любая программа может быть запущена из консоли, если путь до её исполняемого файла (.exe или .bat) указан в переменной PATH.

– Для добавления переменных среды окружения необходимо выполнить следующие действия: вызвать контекстное меню нажав правой кнопкой на эконке Мой Компьютер -> Свойства -> Дополнительные параметры системы -> Переменные среды…» (Рис. 3)

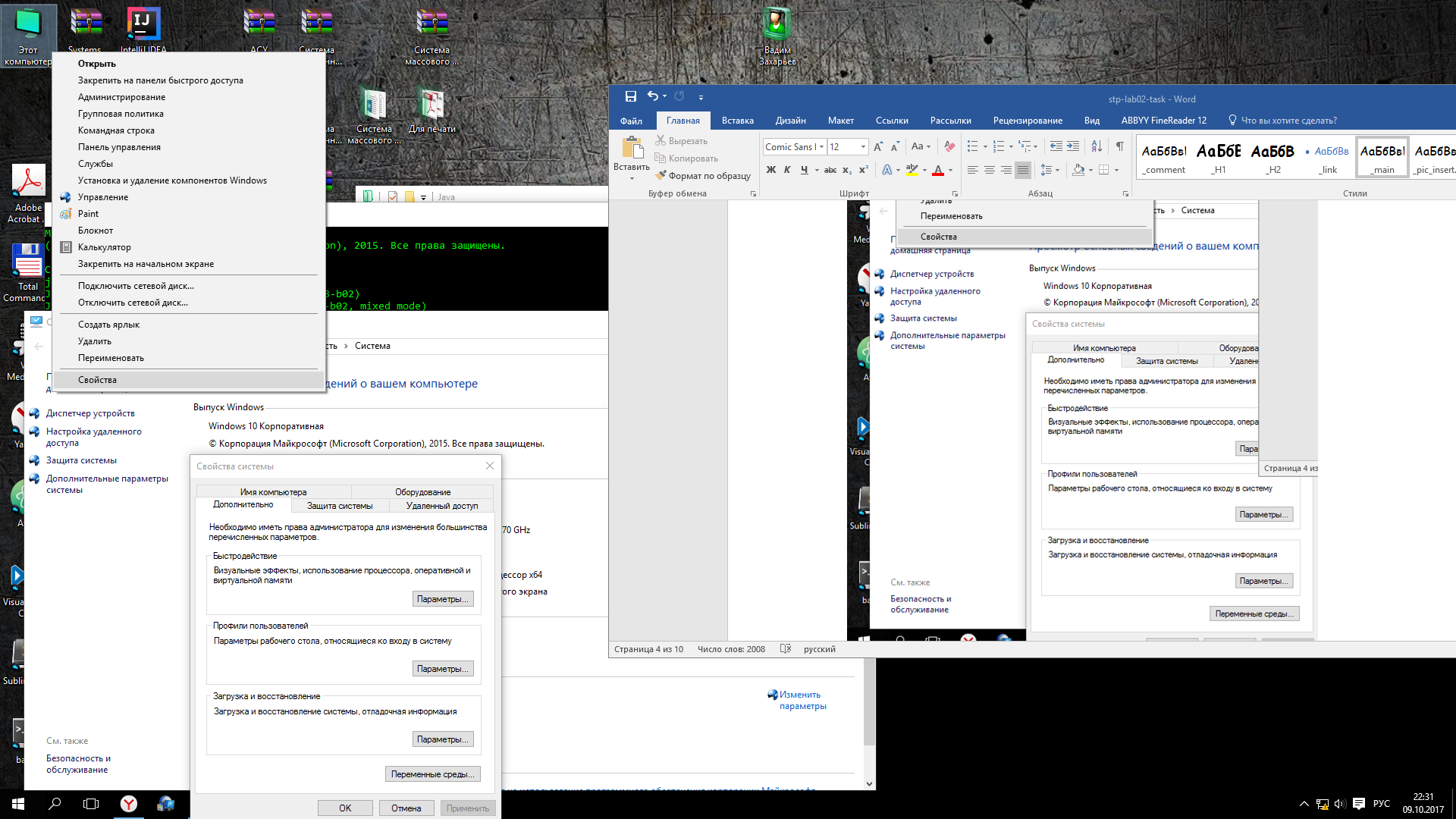


Рис. 3

– Создайте системные переменные JAVA\_HOME, JRE\_HOME, M2\_HOME, CATALINA\_HOME, указав пути к соответствующим директориям (Рис. 4)

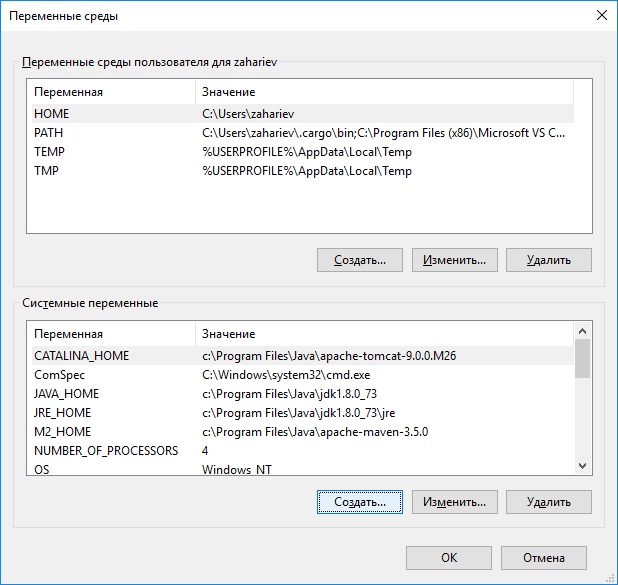
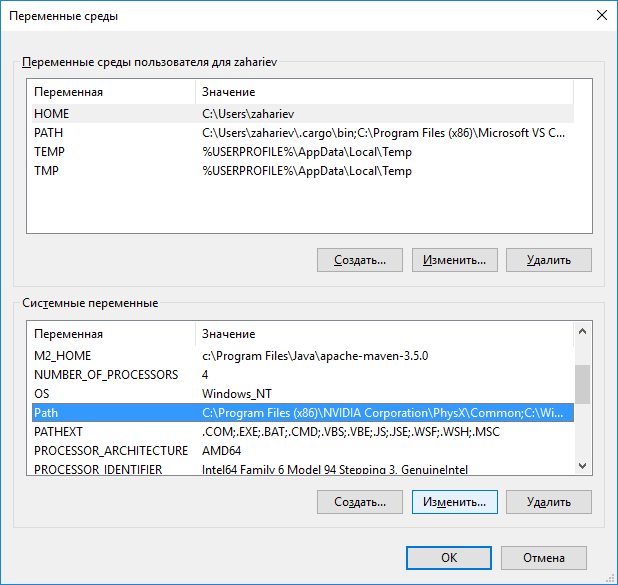


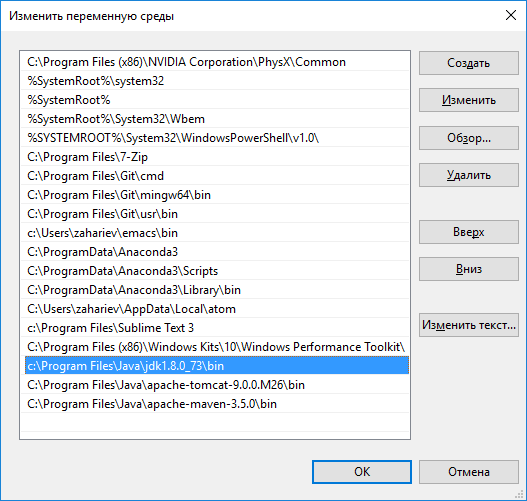
Рис. 4

*Примечание! Обратите внимание, что пути указаны без «слэша» на конце!*

*Т.е. не «С:\Program Files\Java\apache-tomcat-9.0.0.M26\», а «С:\Program Files\Java\apache-tomcat-9.0.0.M26». Будьте осторожны при копировании пути из Total Commander-a!*

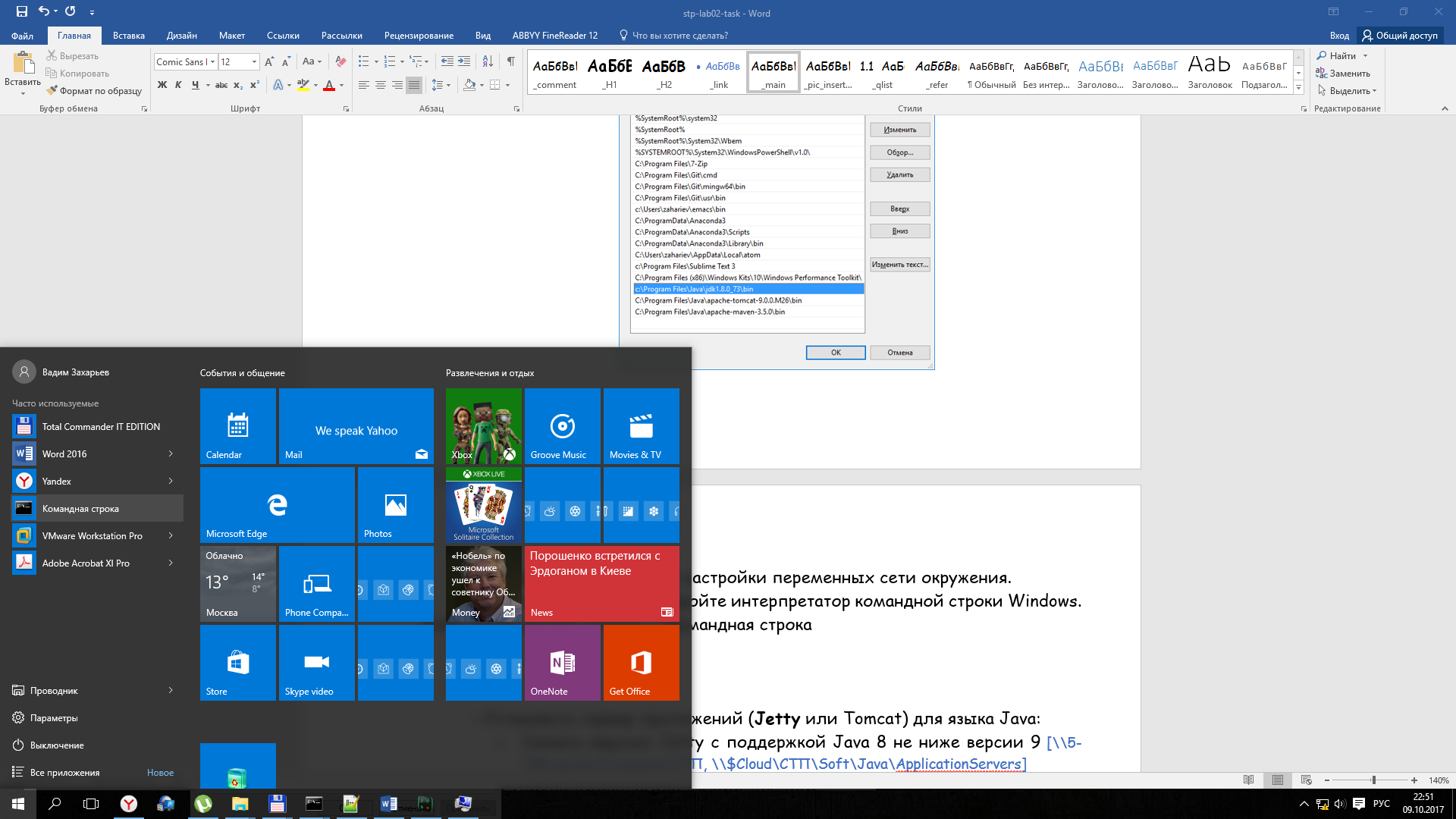
– Добавьте пути к исполняемым файлам в системную переменную PATH. Для этого необходимо выбрать переменную PATH -> нажать “Изменить”.

Добавьте в конец списка строк пути к советующим директориям куда установлены JDK, Tomcat и Maven.

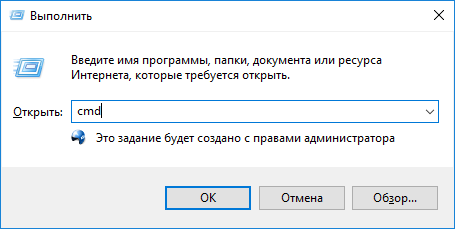


* 1. Проверьте правленость настройки переменных сети окружения.

Для этого выполните откройте интерпретатор командной строки Windows. Либо нажав кнопку «Пуск» -> Командная строка

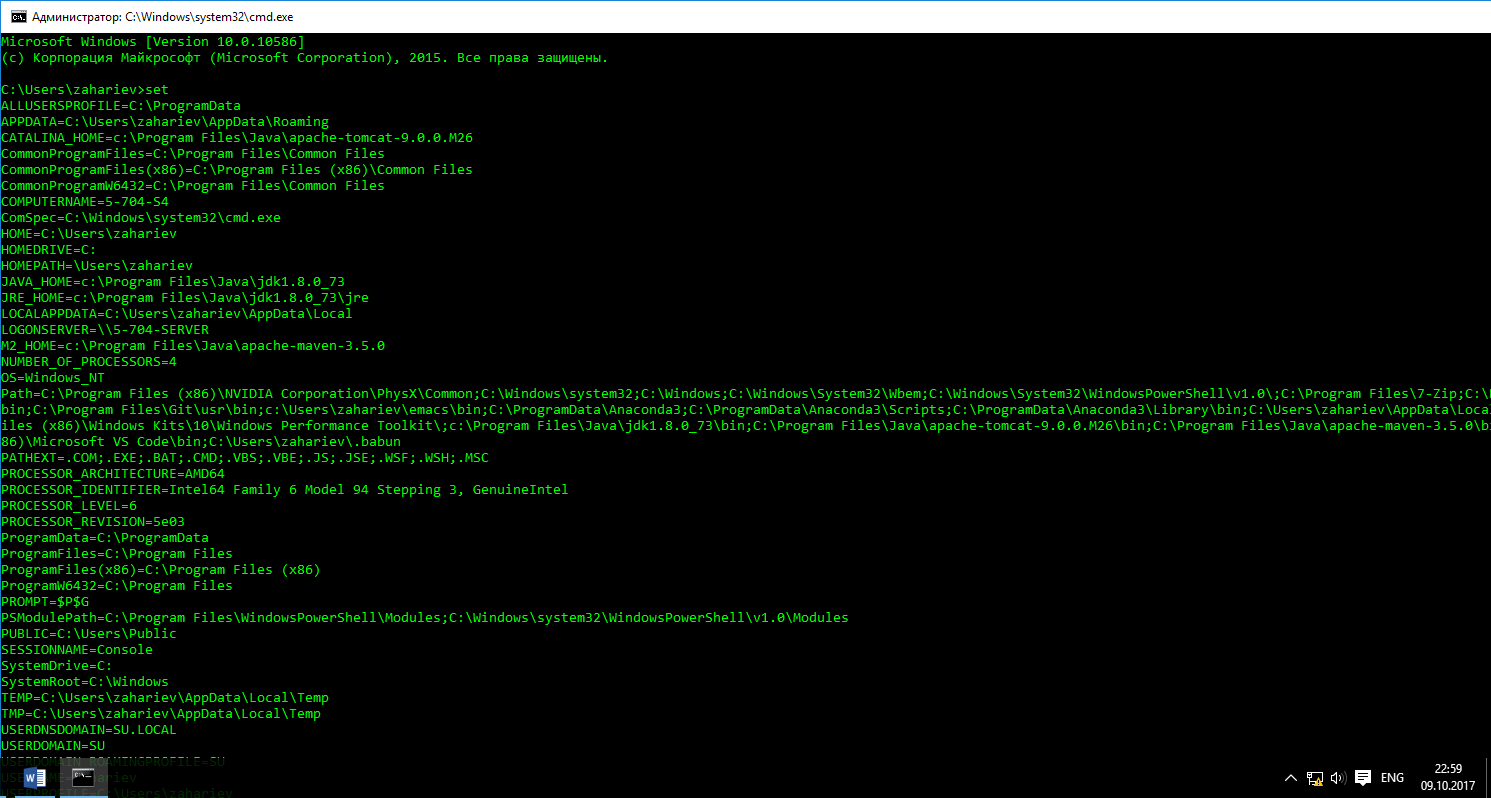


либо посредствам сочетания клавиш «Win+R» вызав меню “Выполнить” и ввести команду «cmd»

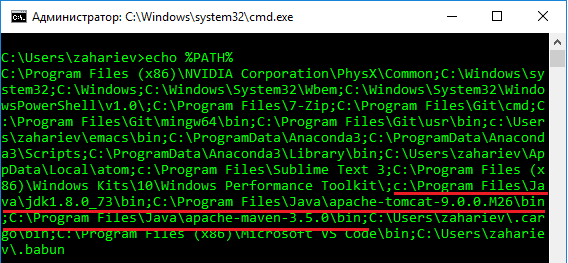


откроется консоль.

Вначале проверим что переменные окружения JAVA\_HOME, JRE\_HOME, M2\_HOME, CATALINA\_HOME установлены. Для этого выполним команду “> set“, которая предназнчена для утсановки переменных. Однако без параметров она просто возвращает их список.

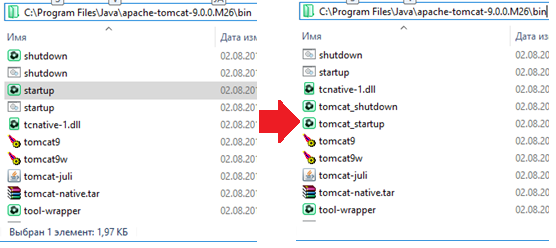


Проверим значение переменной среды PATH, выполнив “> echo %PATH%”

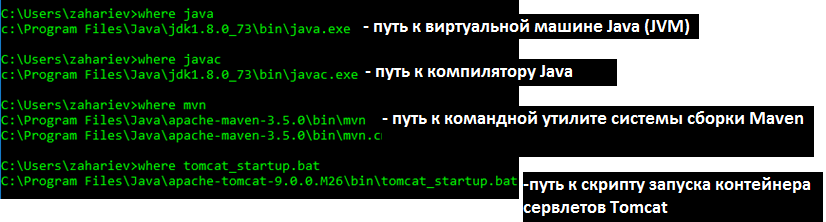


– Проверте доступность исполняемых файлов через консоль с помощью команды «where».

*Примечание! Перед этим шагом зайдите в папку с Tomcat (c:\Program Files\Java\apache-tomcat-9.0.0.M26\bin) и переименуйте файл скрипта, запускающего контейнер сервлетов со* ***startup.bat*** *и* ***shutdown.bat*** *в* ***tomcat\_stratup.bat*** *и* ***tomcat\_startup.bat*** *соответственно. Это необходимо для того чтобы избежать возможны ошибок при вызове скриптов из консоли, когда интерпретатор командной строки не путал их с другими скриптами, которые также могут быть доступны через переменную PATH. Имя strartup и shutdown в данном случае являются весьма общими. Поскольку в дальнейшем в системе может быть установлена, например, СУБД, которая также запускается с использованием скрипта с подобным именем.*



– Выполните поочередно команду where c названием исполняемых файлов для установления возможности их запуска из командной строки. Команда where возвращает полный путь к исполняемому файлу, если он доступен в системе.



*Внимание! Убедитесь что к файлам java и javac команда where возвращает только один путь. Если например к команде видут два пути, например*

*> where java*

*C:\Program Files\Oracle\jdk\java (убрать из переменной PATH)*

*C:\Windows\System32\java (переименовать в папке файл в java.bak )*

С:\Program Files\jdk1.8.0\_73\bin\java.exe (ПРАВИЛЬНЫЙ ПУТЬ!)

*Должна остаться ссылка только на один файл c правильным путем!*

* 1. Скачать и установить интегрированную среду разработки Intellij IDEA (Eclipse, NetBeans) для разработки приложений на Java.

**Интегрированная среда разработки** (Integrated development environment — IDE) комплекс программных средств, используемый программистами для разработки программного обеспечения.

Среда разработки включает в себя:

* текстовый редактор,
* компилятор и/или интерпретатор,
* средства автоматизации сборки,
* отладчик.

Иногда содержит также средства для интеграции с системами управления версиями и разнообразные инструменты для упрощения конструирования графического интерфейса пользователя. Многие современные среды разработки также включают браузер классов, инспектор объектов и диаграмму иерархии классов — для использования при объектно-ориентированной разработке ПО. ИСР обычно предназначены для нескольких языков программирования — такие как IntelliJ IDEA, NetBeans, Eclipse, Qt Creator, Geany, Embarcadero RAD Studio, Code::Blocks, Xcode или Microsoft Visual Studio, но есть и IDE для одного определённого языка программирования — как, например, Visual Basic, Delphi, Dev-C++.

– Скачать в учебной аудитории можно по ссылке:

[\\5-704-SERVER\pub\Programming\Java\IDE\IntellijIdea\](file:///\\5-704-SERVER\pub\Programming\Java\IDE\IntellijIdea\%20)

дома с официального сайта:

<https://jetbrains.ru/products/idea/>

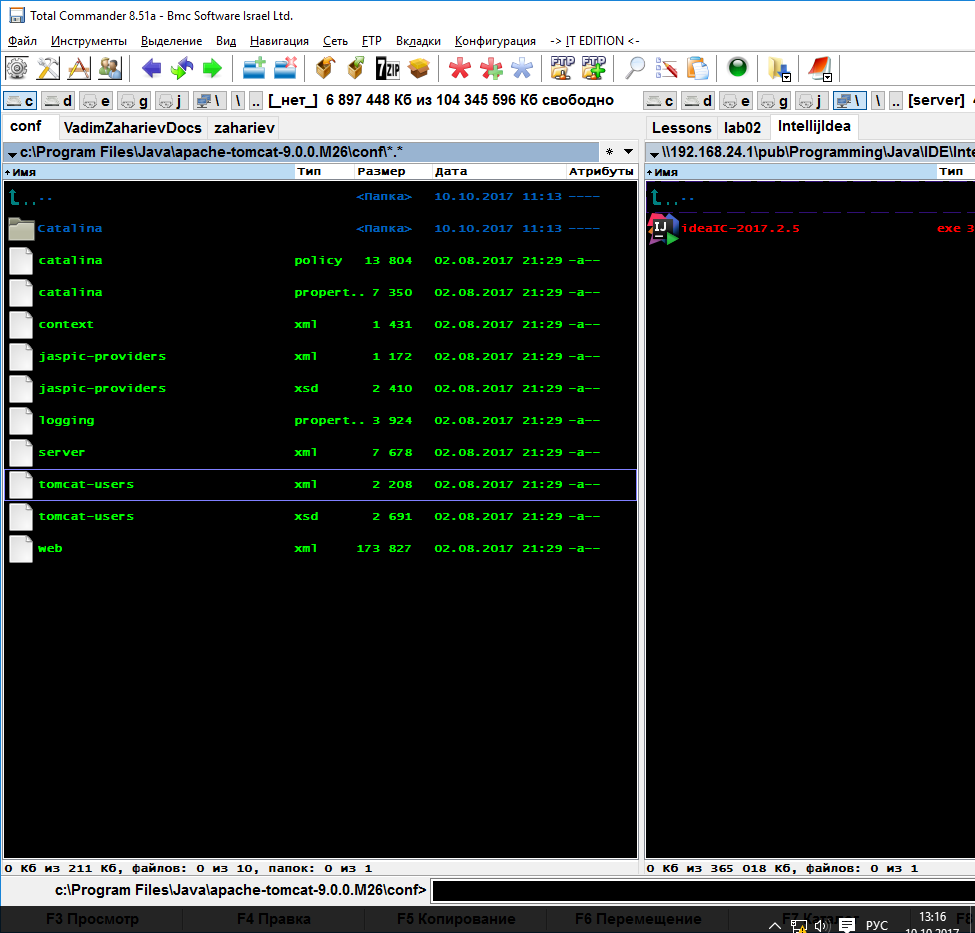
или по запросу «intellij idea community edition» в поисковой системе.

*Примечание! В примере используется версия «IntelliJ IDEA* ***Community Edition*** *2017.2.5». Если вы решили воспользоваться альтернативной IDE все последующие действия вам необходимо будет выполнять с учетом возможностей, пользовательского интерфейса и элементов управления выбранной среды разработки.*

-

* 1. Настроить группы пользователей для обеспечения доступа к контейнеру сервлетов Tomcat через веб-интерфейс.

– Для этого в конце файла «**tomcat-users.xml**» в директории : С:\Program Files\Java\apache-tomcat-9.0.0.M26\conf

**

добавить следующие строчки внутри элемента <tomcat-users> :

<**tomcat-users>**

……

<role rolename="manager-script"/>

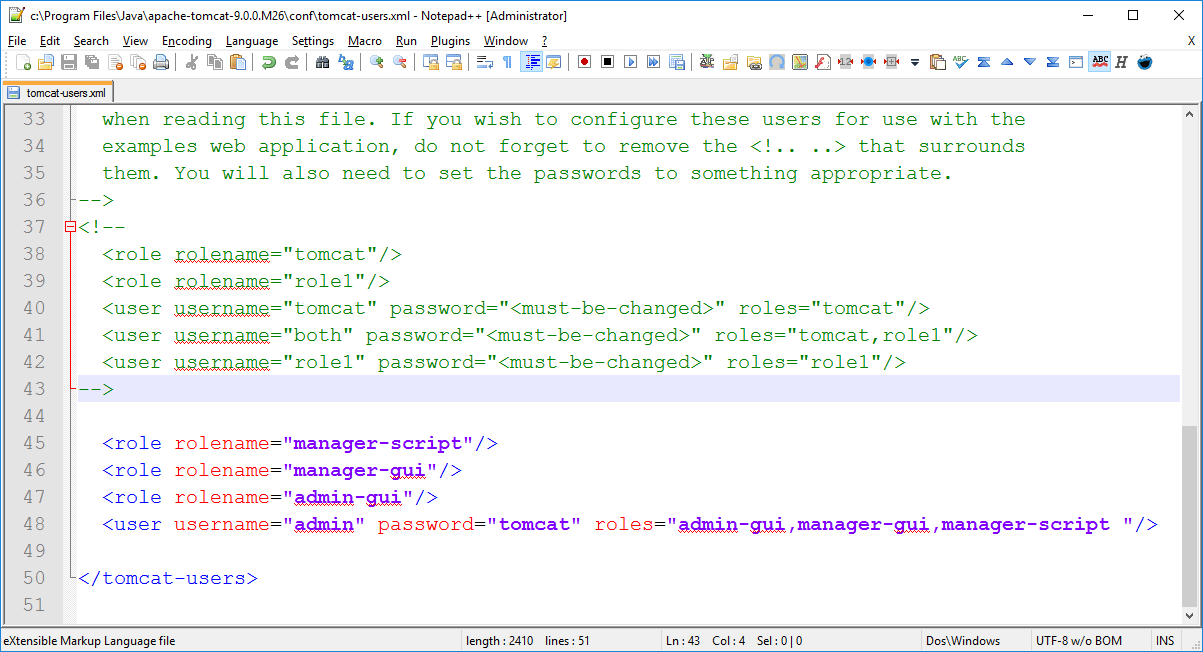
<role rolename="manager-gui"/>

<role rolename="admin-gui"/>

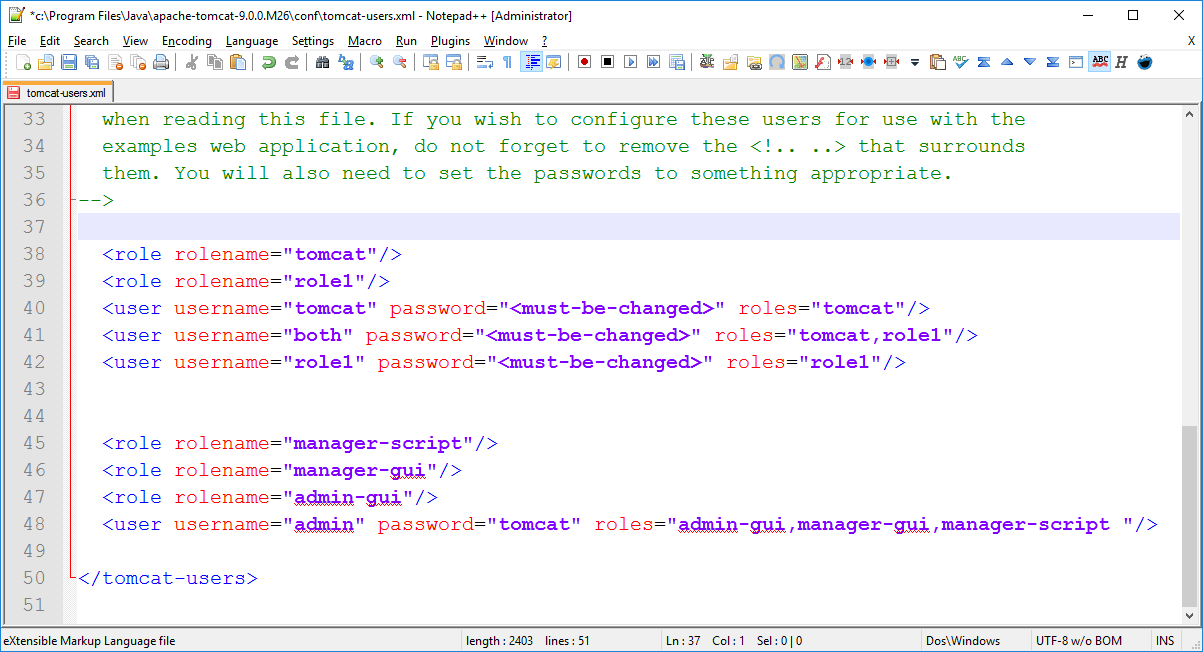
<user username="admin" password="tomcat" roles="admin-gui,manager-gui,manager-script "/>

….

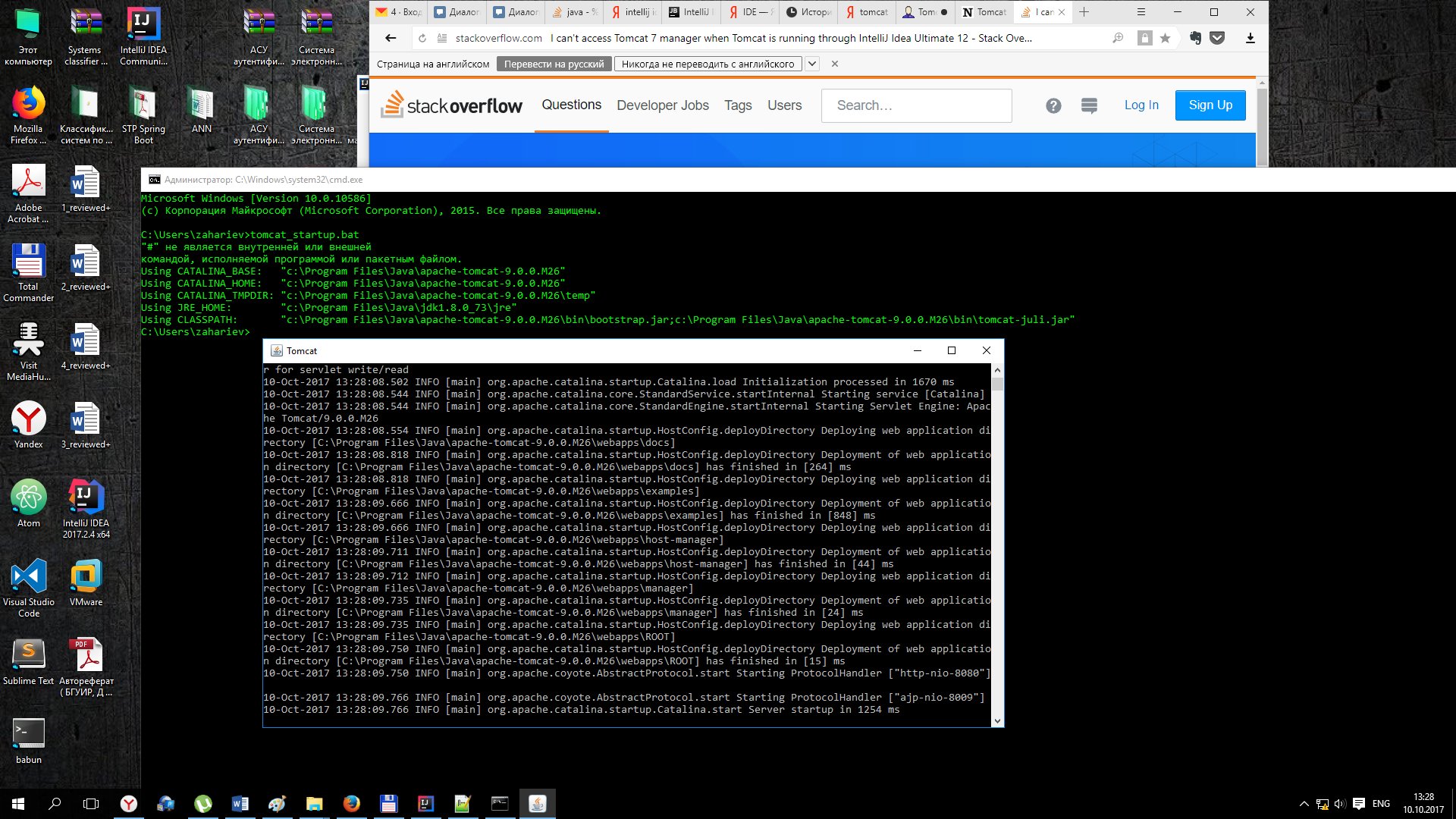
**</tomcat-users>**



Примечание! По желанию можно также расскоментировать блок со стандартными ролями Tomcat. Более подробно про роли доступа к серверу читайте в документации.

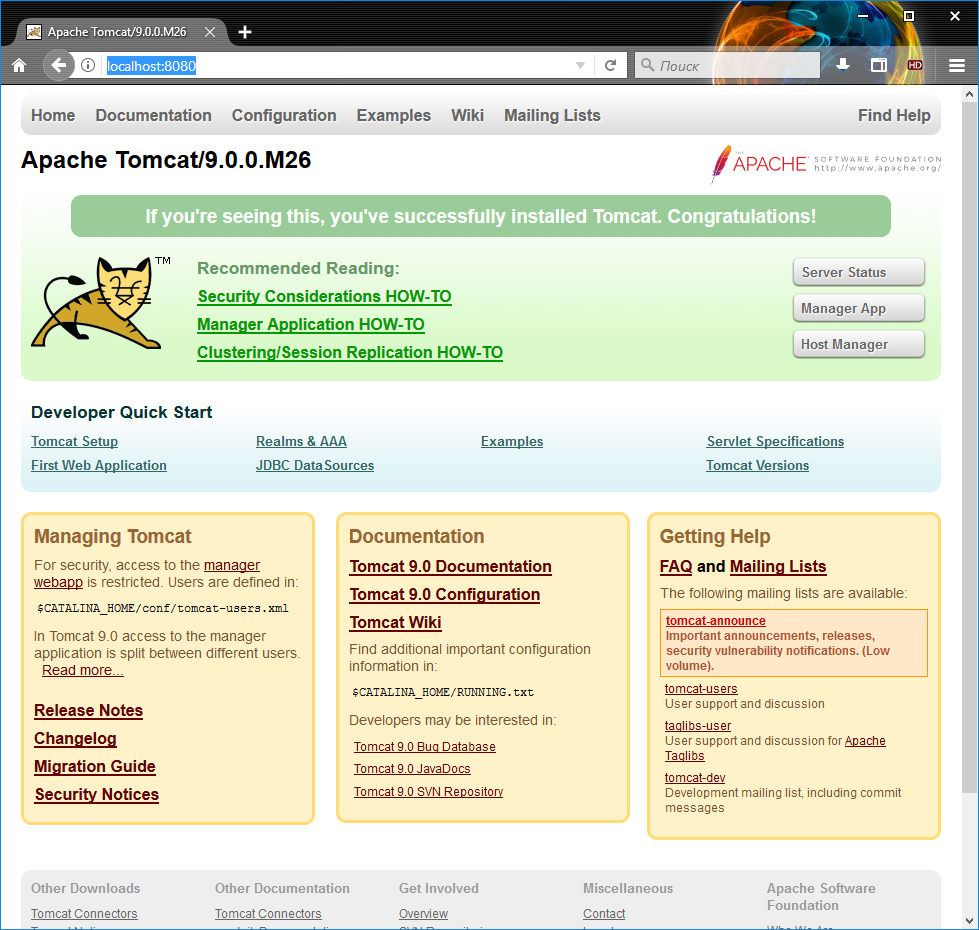


– Запустить Tomcat командной строки, командной tomcat\_startup.h.

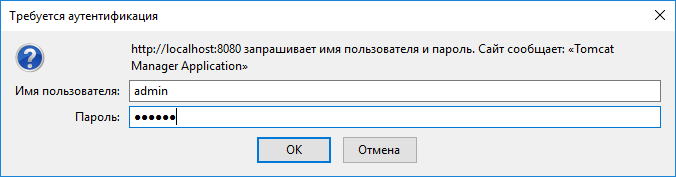


– Открыть браузер и перейти по адресу локально работающего модуля веб-сервера контейнера сервлетов Tomcat: <http://localhost:8080/>

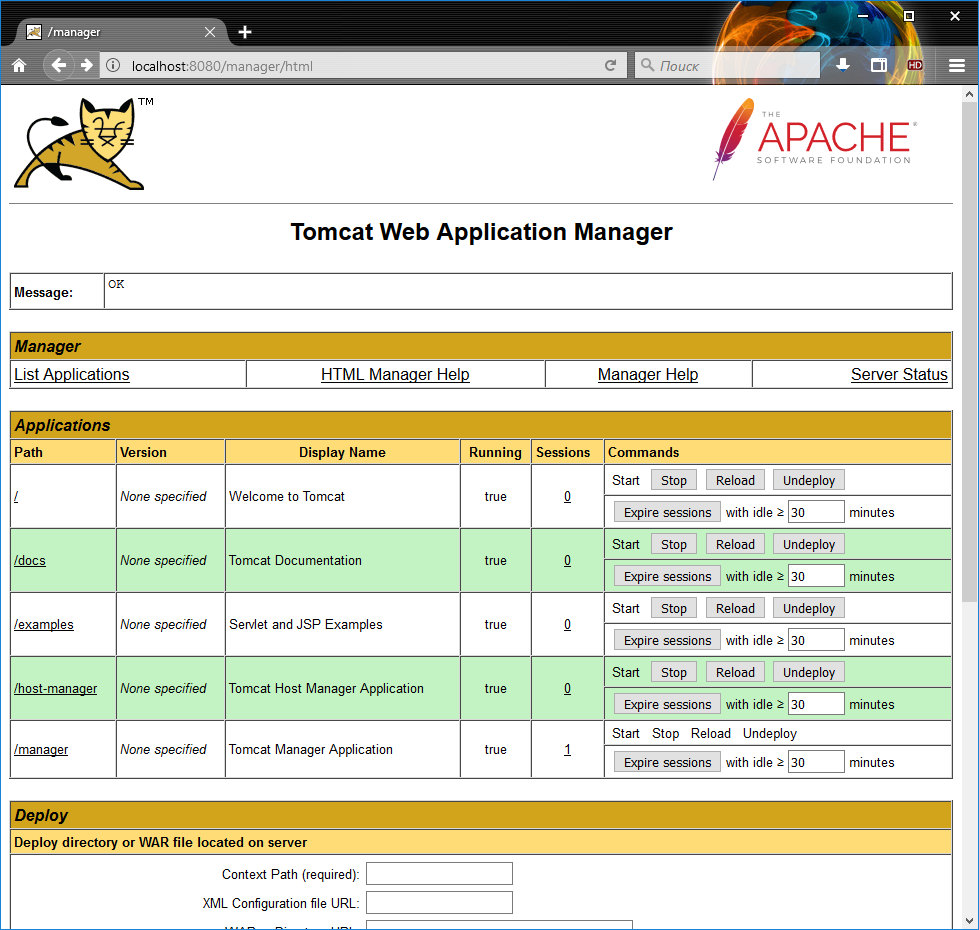
Должно появиться страница указывающая на то что сервер функционирует.



1.8 Проверить доступ к менеджеру развертывания приложений, нажав на кнопку “Manager App” и введя логин и пароль пользователя прописанного в файле **tomcat-users.xml** на предыдущем этапе. В нашем случае logn: admin, pwd: tomcat.



Вы должны увидеть панель управления сервером.



Можете немного расслабиться, отдохнуть, выпить чашечку кофе и посмотреть примеры сервлетов уже развернутых в контейнере сервлетов по адресу: http://localhost:8080/examples/

Поздравляем ваше рабочее окружение готово! Теперь вы можете легко и непринуждённо приступить к разработке веб-приложений на языке Java! ; )

2. Теоретическая часть:

2.1 Ознакомиться с материалами главы 15, стр. 456-461 из книги «Технология разработки Web-приложений» книги И.Н. Блинов, В.С. Романчик, Java. Методы программирования. 2013 г. – Скачать в учебной аудитории можно по ссылке:

[\\5-704-SERVER \Lessons\СТП\Методички\books\](\\\\5-704-SERVER \\Lessons\\СТП\\Методички\\books\\)

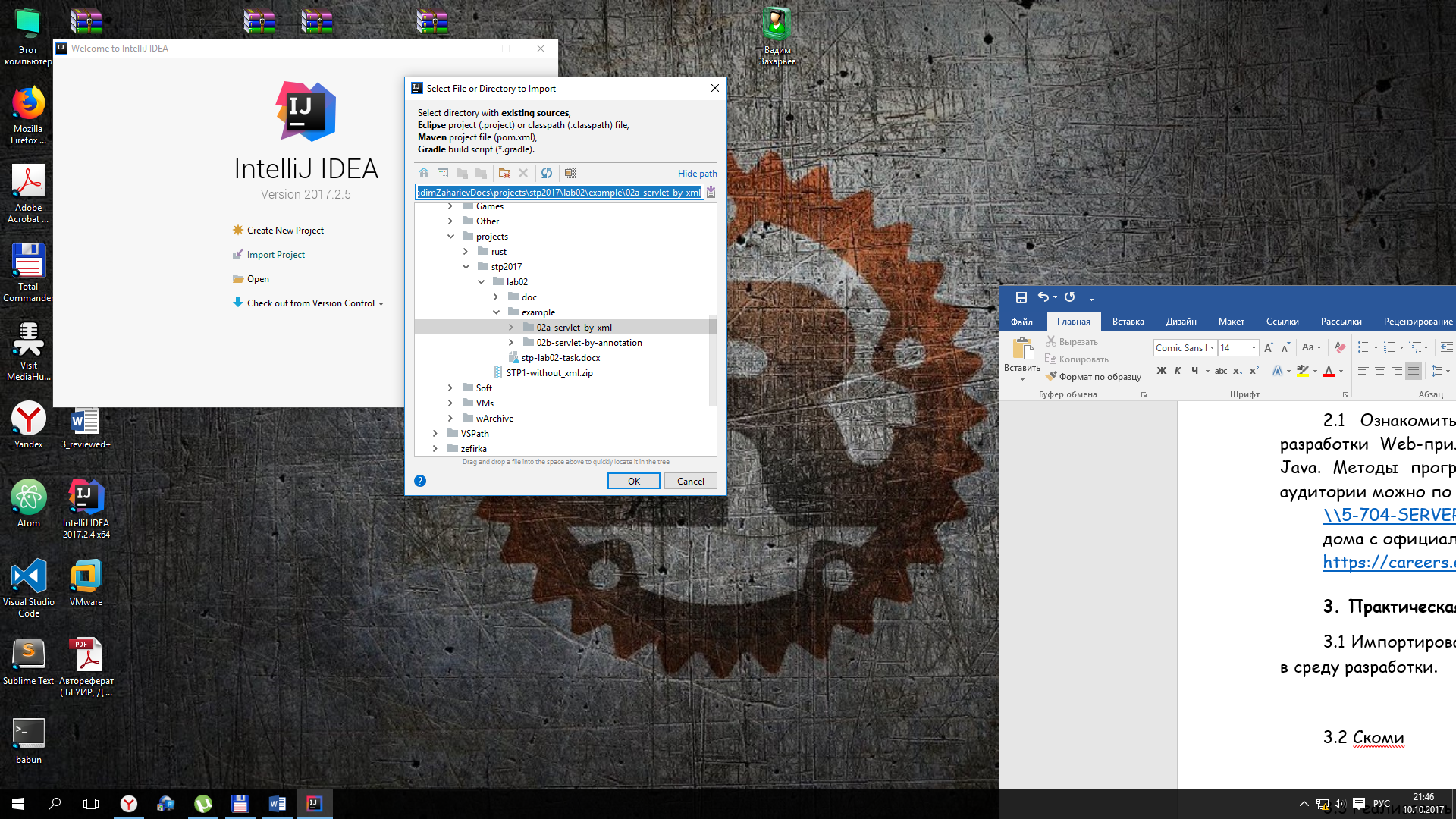
дома с официального сайта:

<https://careers.epam.by/training/books>

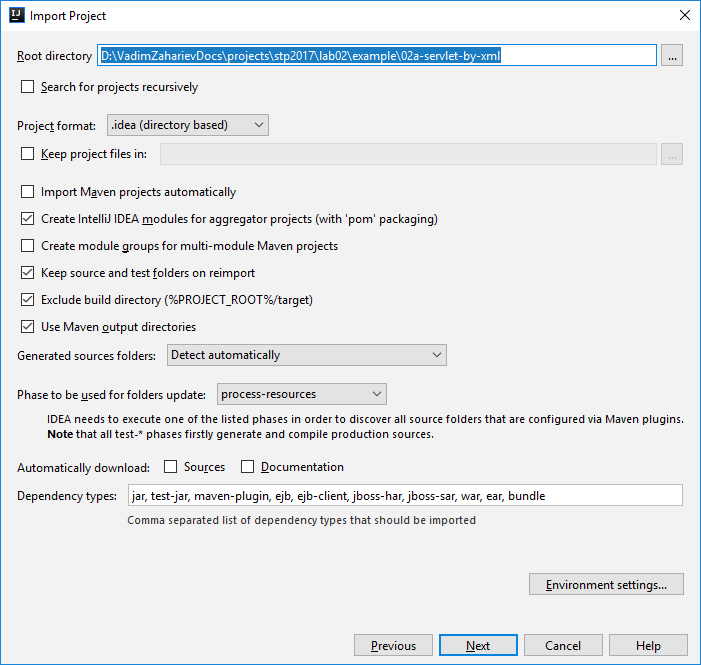
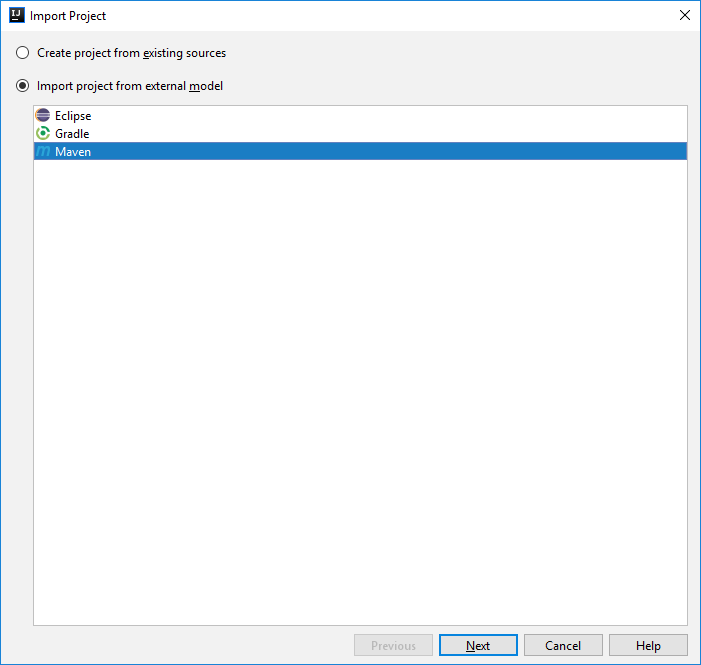
3. Практическая часть:

3.1 Изучить подход к созданию сервлета Java на основе привязки через файл web.xml, и запустить данный сервлет в контейнере сервелатов Tomcat.

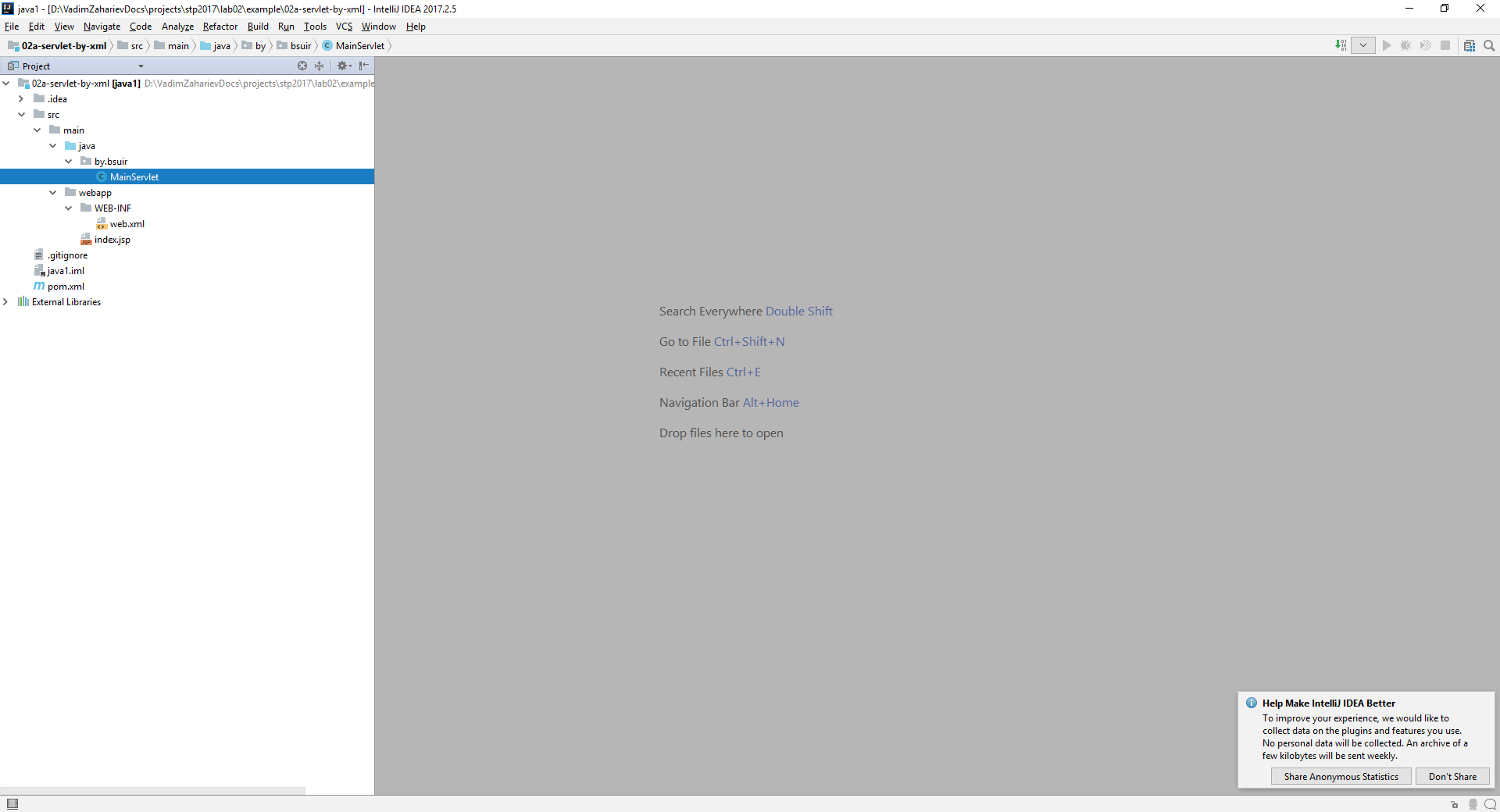
3.2 Импортировать в Intellij IDEA проект «02a-servlet-by-xml» в среду разработки.



В диалоговом окне необходимо указать путь к каталогу «02a-servlet-by-xml» и импортировать проект с использованием Maven на основе файла описания файла pom.xml.

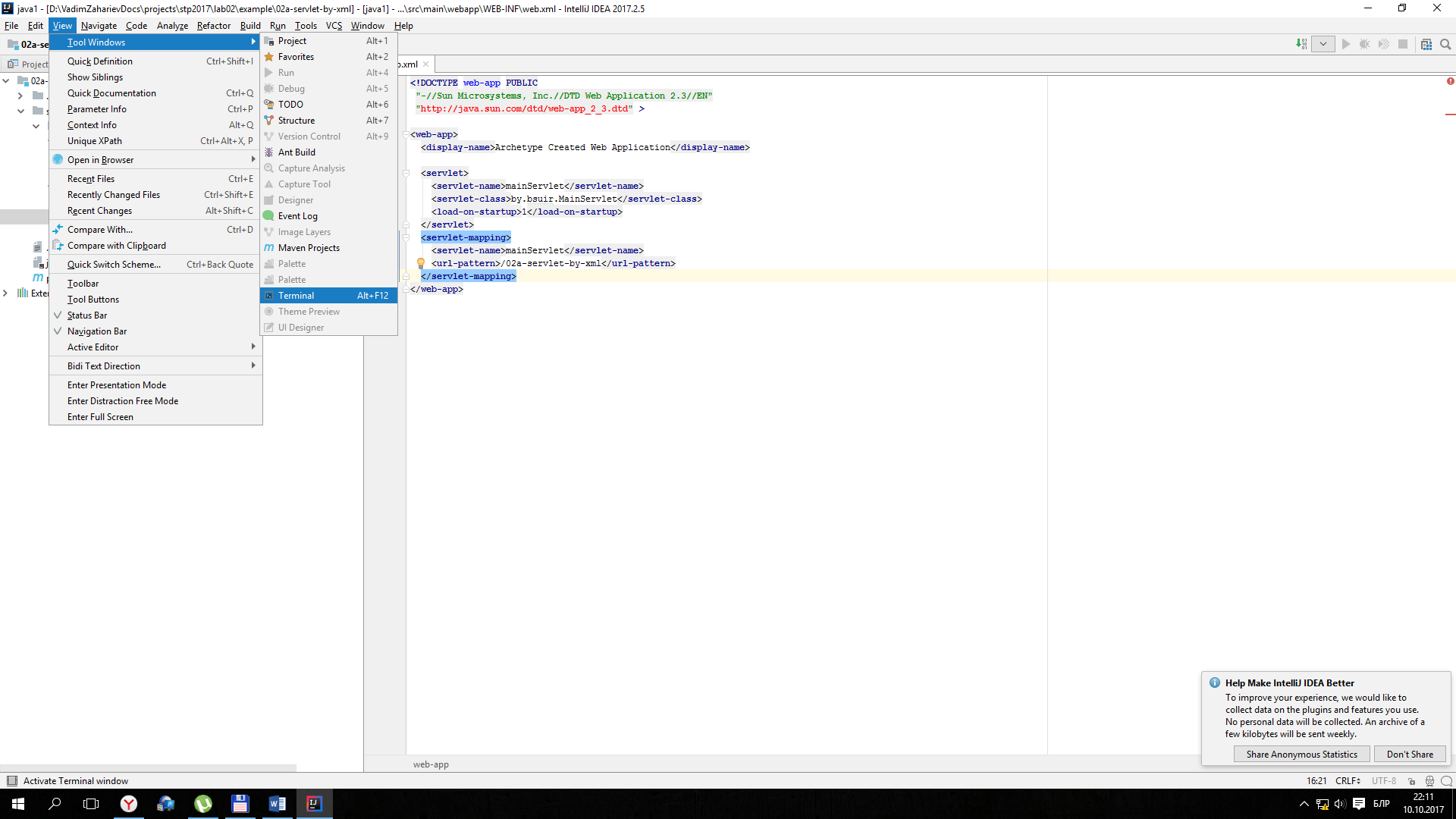


В итоге должно открыться дерево проекта

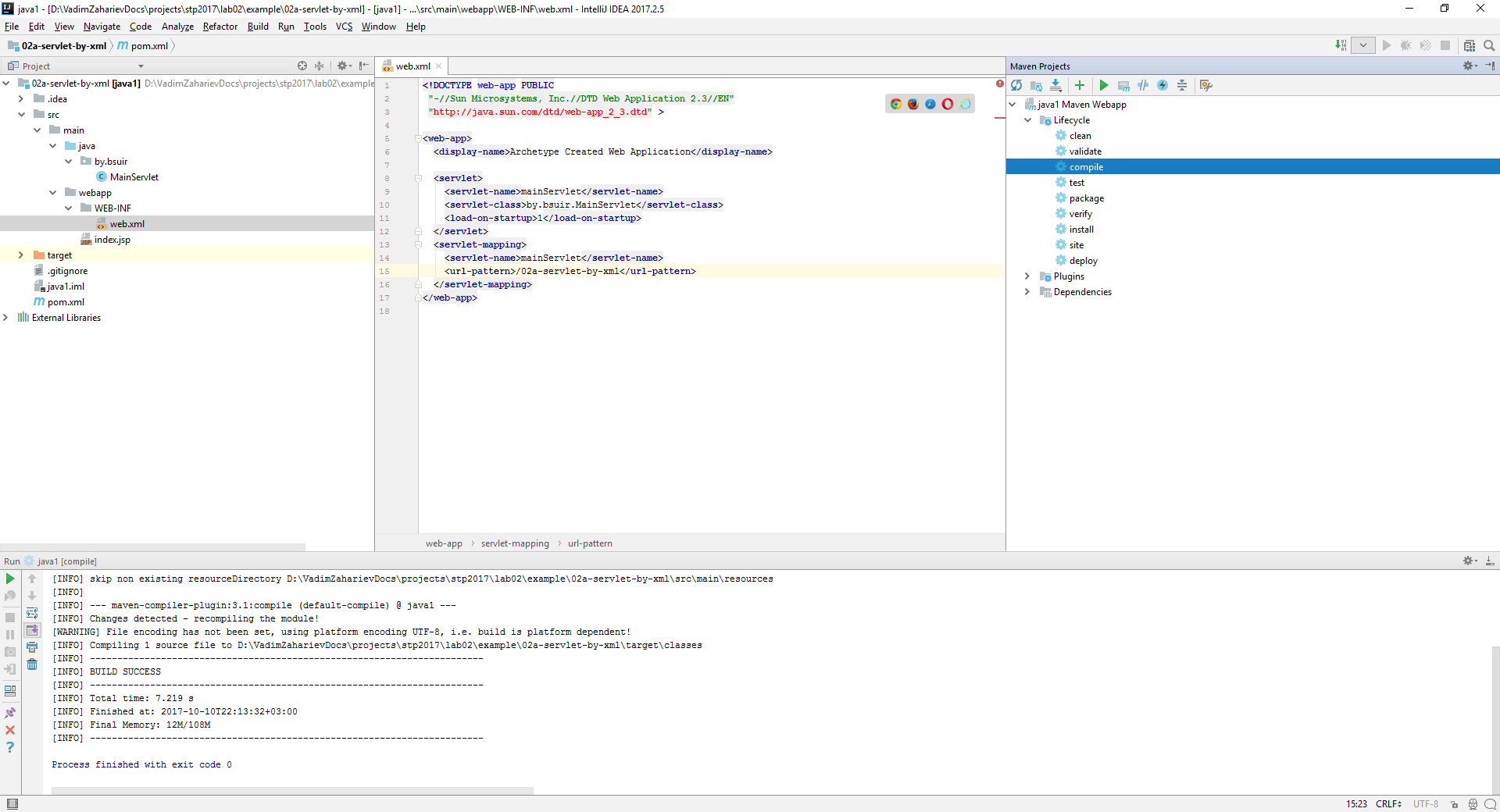


3.3 Изучить класс MainServlet а также файл web.xml для привязки (mapping/маппинга) сервлета к опреденному пути.

3.4 Открыть терминал в Intellij IDEA, перейдя в панели инструментов через пункты View -> Tool Windwos -> Terminal.



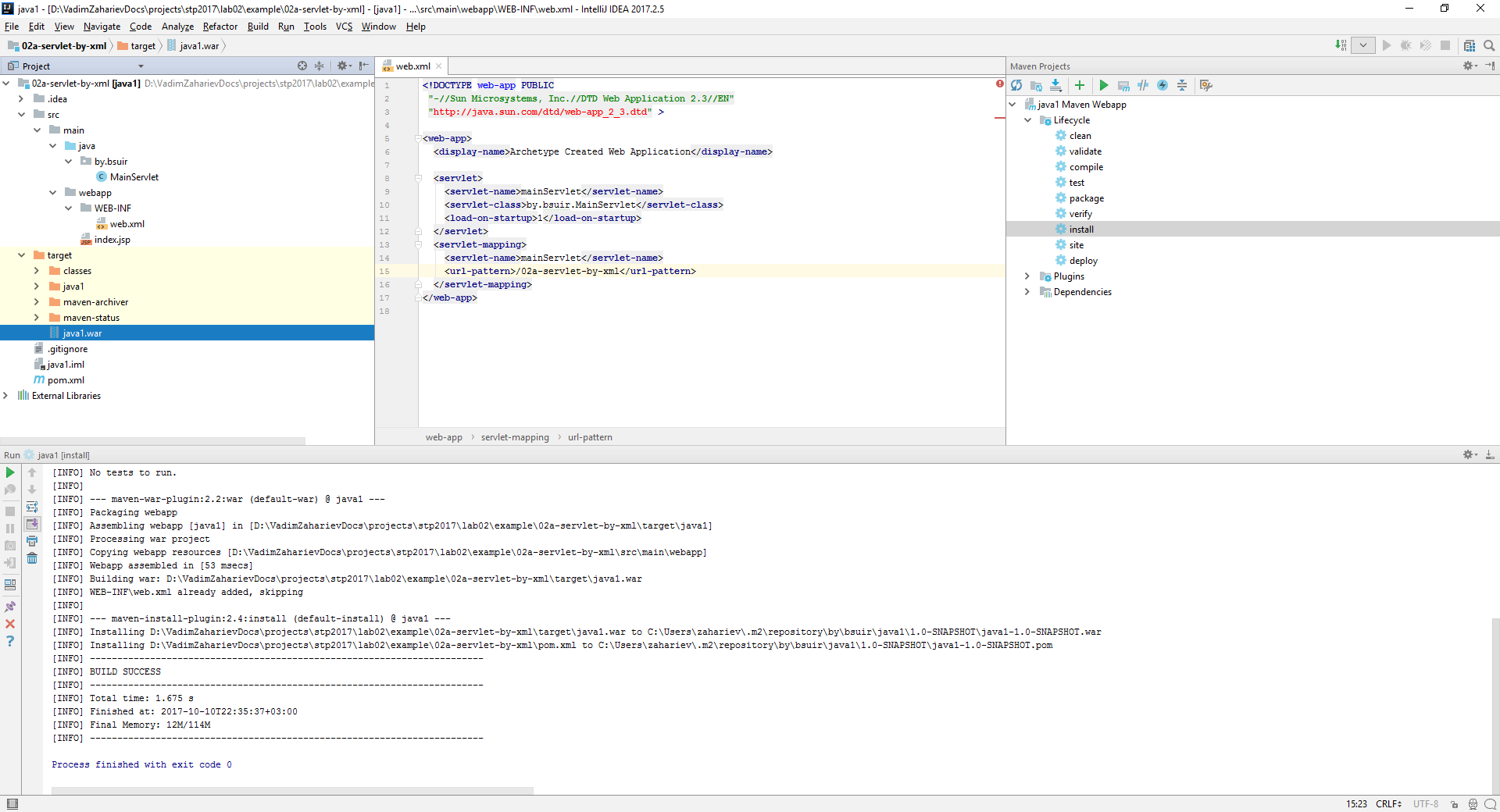
3.5 Открыть панель Maven Project через вкладку View -> Tool Windwos -> Maven Project.



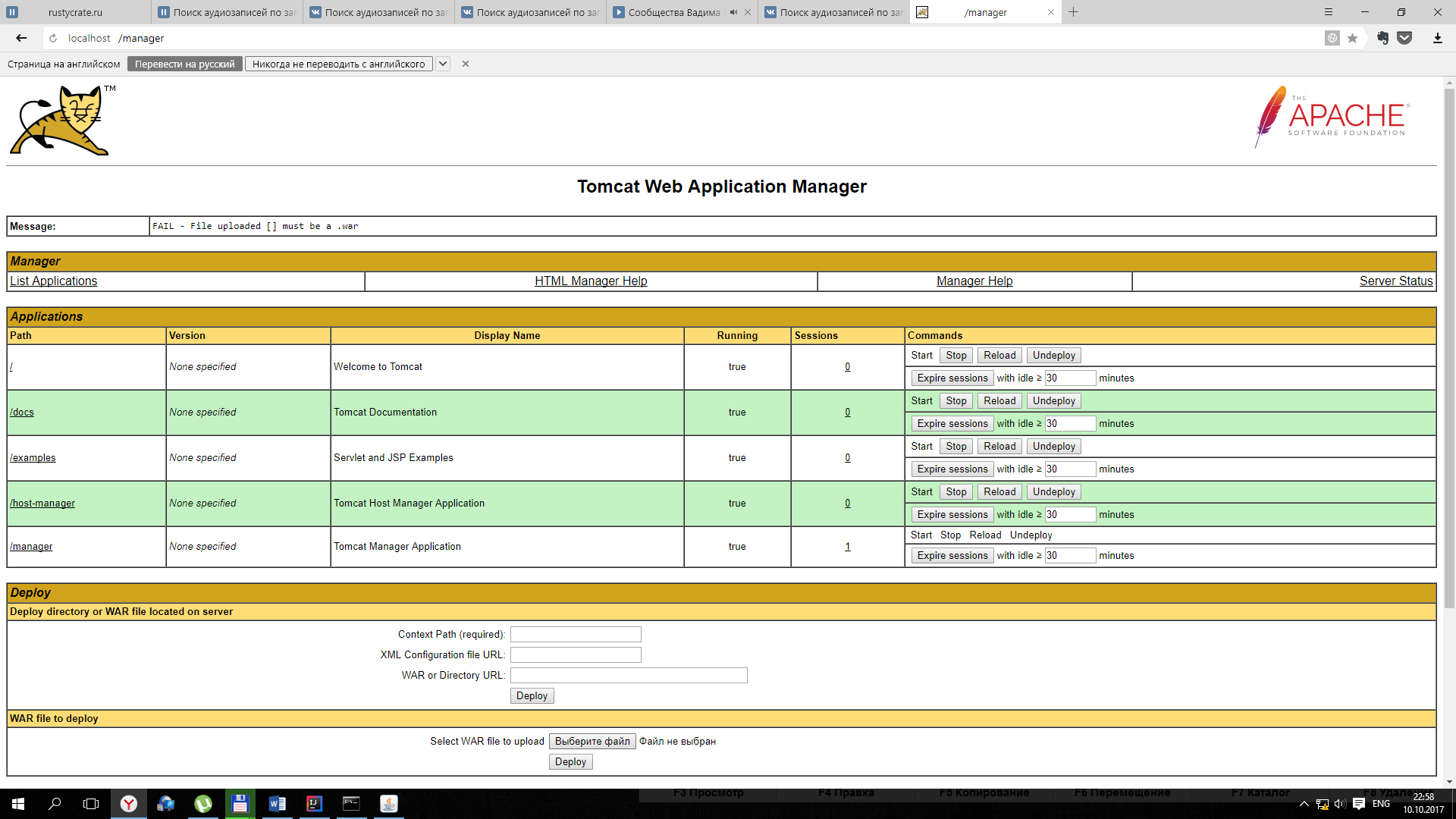
3.6 Выполнить компиляцию проекта с помощью команды compile, далее упаковку с помощью команды package.

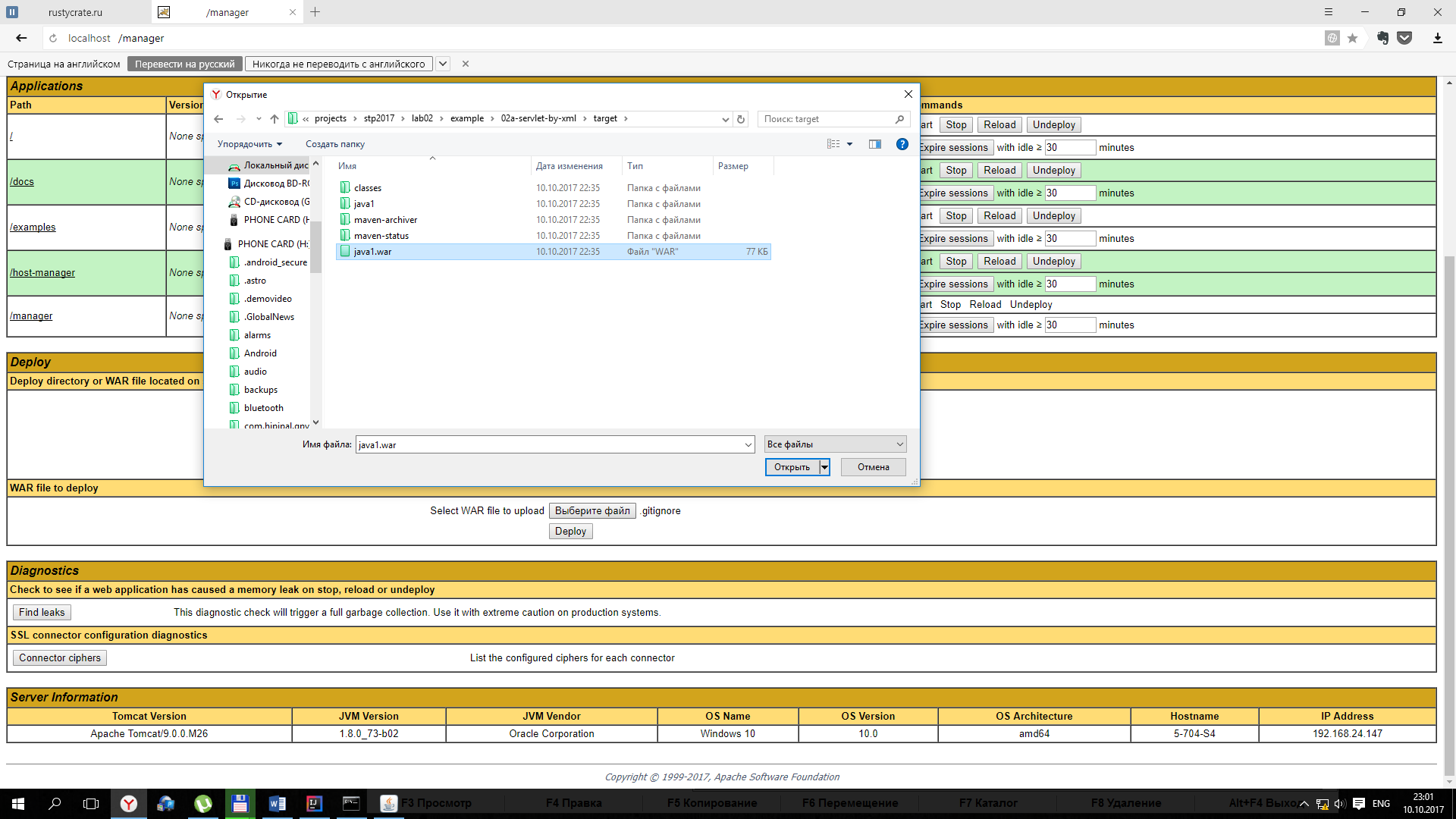


В дереве проекта появиться папка target, в которой будет находится скомпилированный и упакованный файл проекта \*.war.

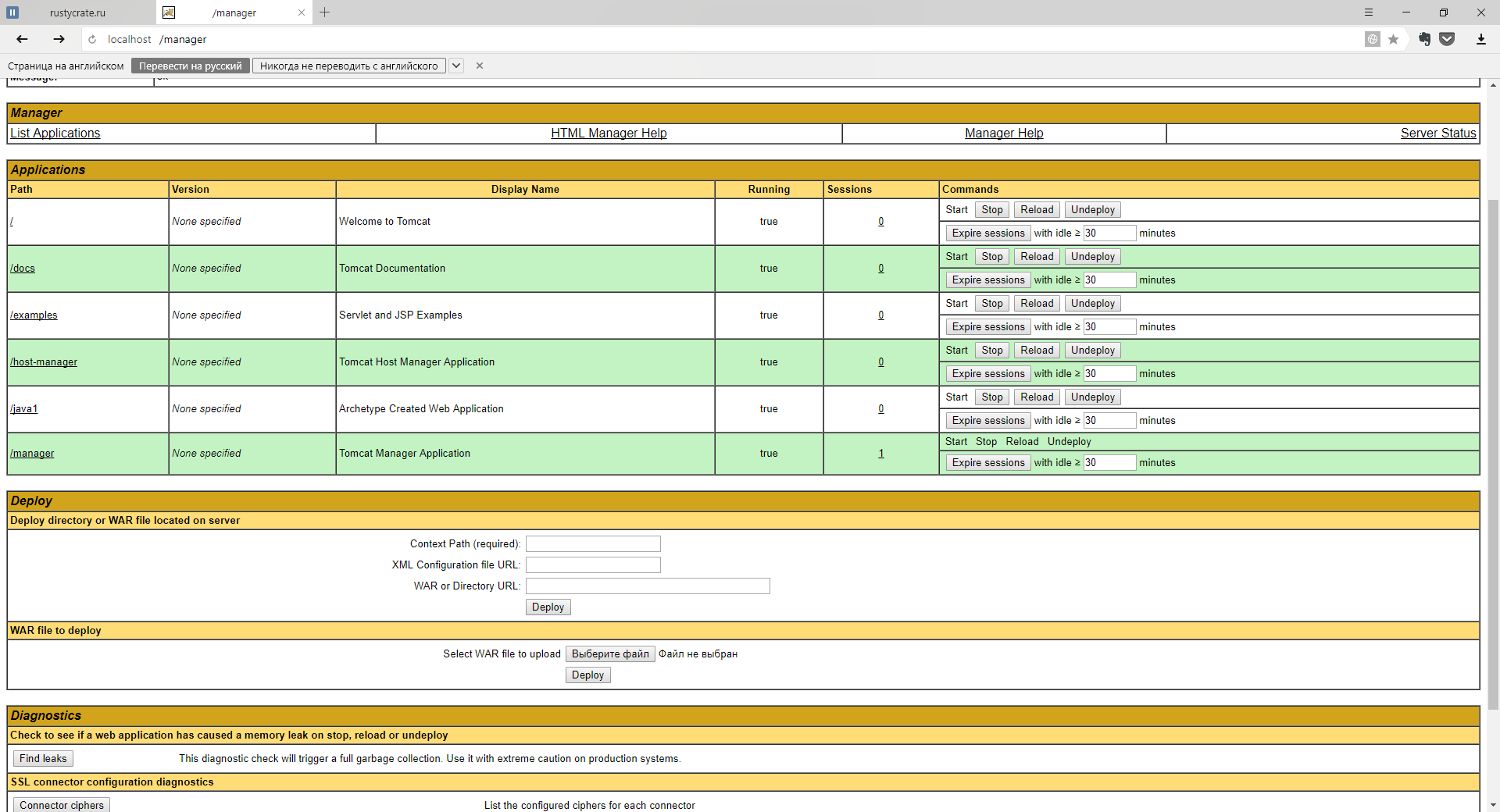


3.7 Запустить Tomcat, выполнив команду tomcat\_startup.bat войти администраторскую панель и развернуть файл архива, содержащего сервлет java1.wav в контейнере сервлетов, с помощью кнопки Deploy. Указать путь к архиву сервлета java1.wav.

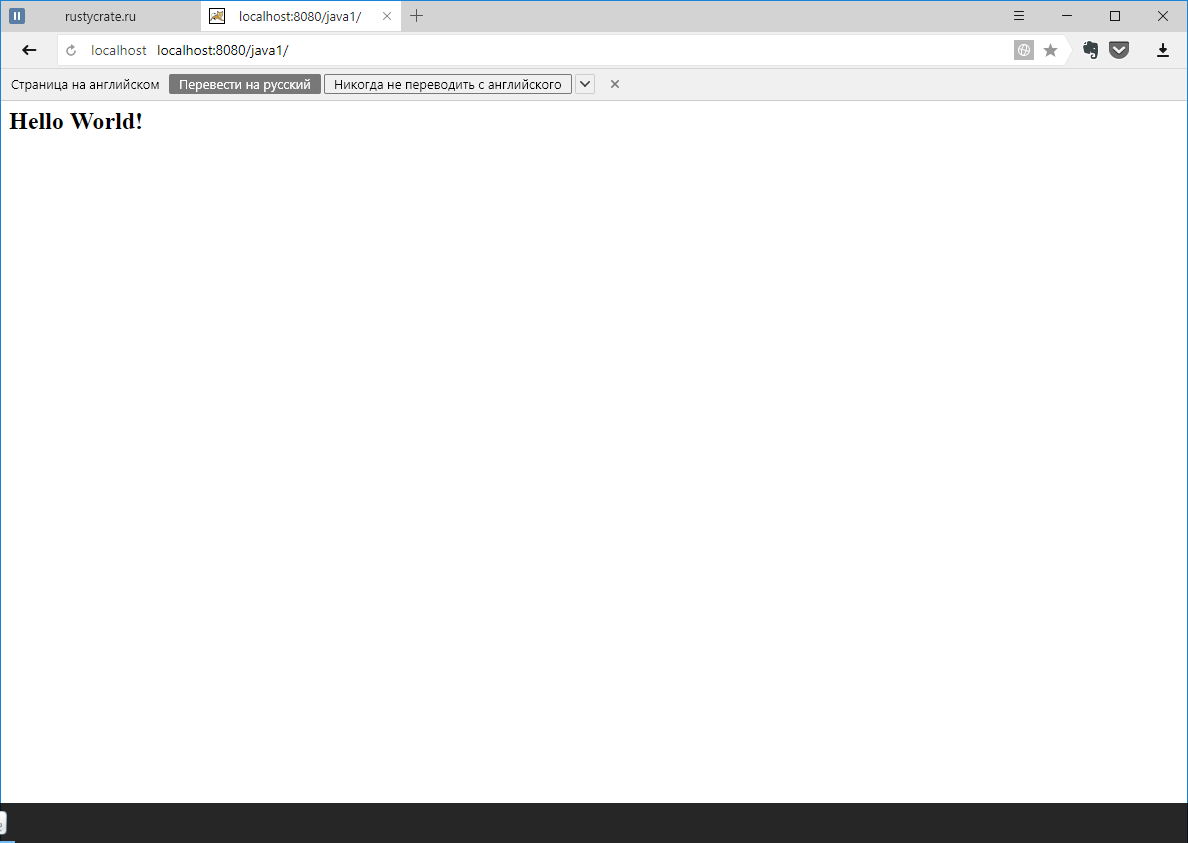




3.8 Запустить Tomcat, выполнив команду tomcat\_startup.bat войти администраторскую панель и развернуть файл архива, содержащего сервлет java1.wav в контейнере сервлетов, с помощью кнопки Deploy. Указать путь к архиву сервлета java1.war.



3.9 В браузере обратиться по URL вызвать сервлет.



3.10 Импортировать в Intellij IDEA проект «02b-servlet-by-annotation» разобраться как производиться маппинг сервлета. Скомпилировать и развернуть (deploy) на сервере данный сервлет.

3.11. Добавить данные сервлеты в ваш Git репозиторий в различные ветки проекта. Ветки назвать в соответствии с названием задания «02a-servlet-by-annotation» и «02b-servlet-by-xml».

3.12 Написать свой собственный сервлет, который в виде сообщения ответа будет возвращать индексную страницу вашего сайта, разработанную в лабораторной №1.