**Міністерство освіти і науки України**

**Харківський національний університет ім. В. Н. Каразіна**

**Факультет комп’ютерних наук**

**Кафедра безпеки iнформацiйних систем i технологiй**

**Лабораторна робота № 4**

*з навчальної дисципліни*

«**Математичні методи та технології тестування і верифікації програмного забезпечення**»

***Виконала:***

студентка групи КБ-22

Рогоза П. В.

***Перевірила:***

Мелкозьорова О. М.

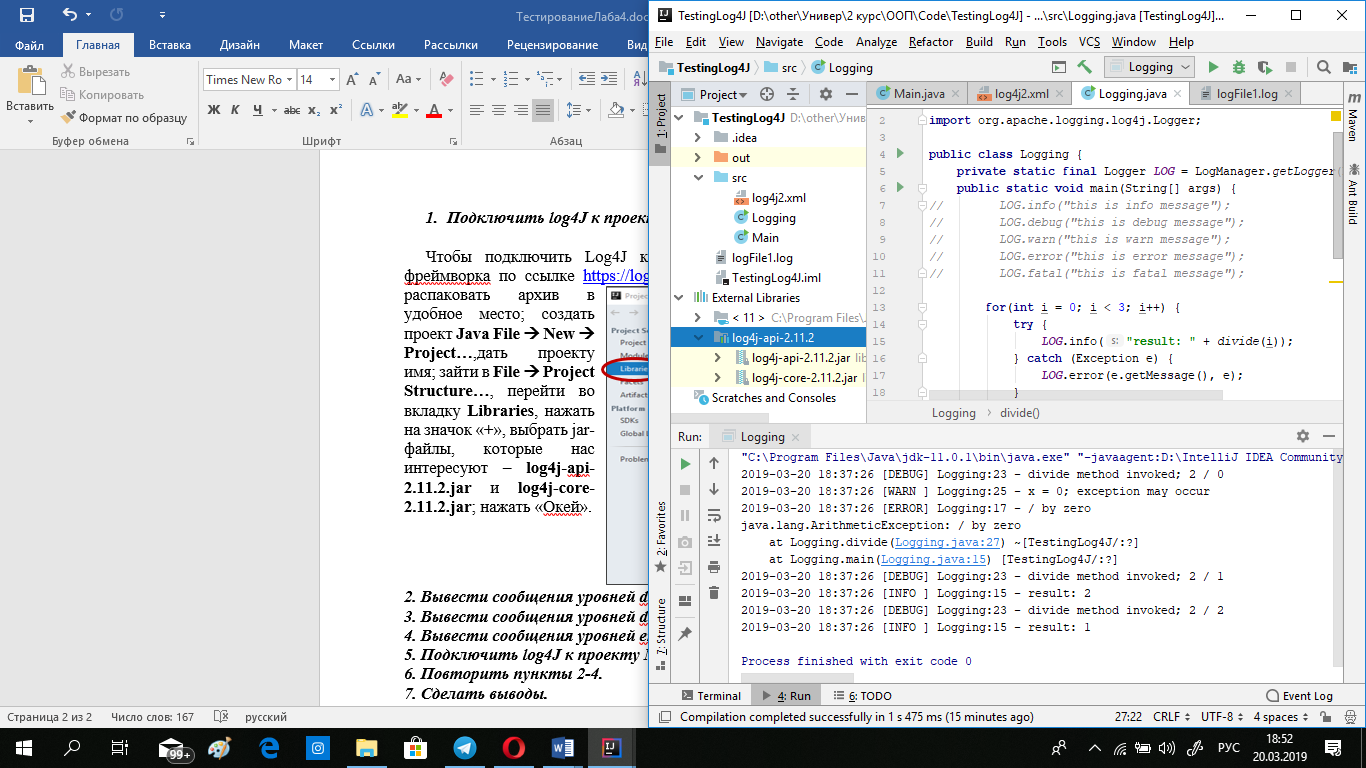
Харків – 2019 р.

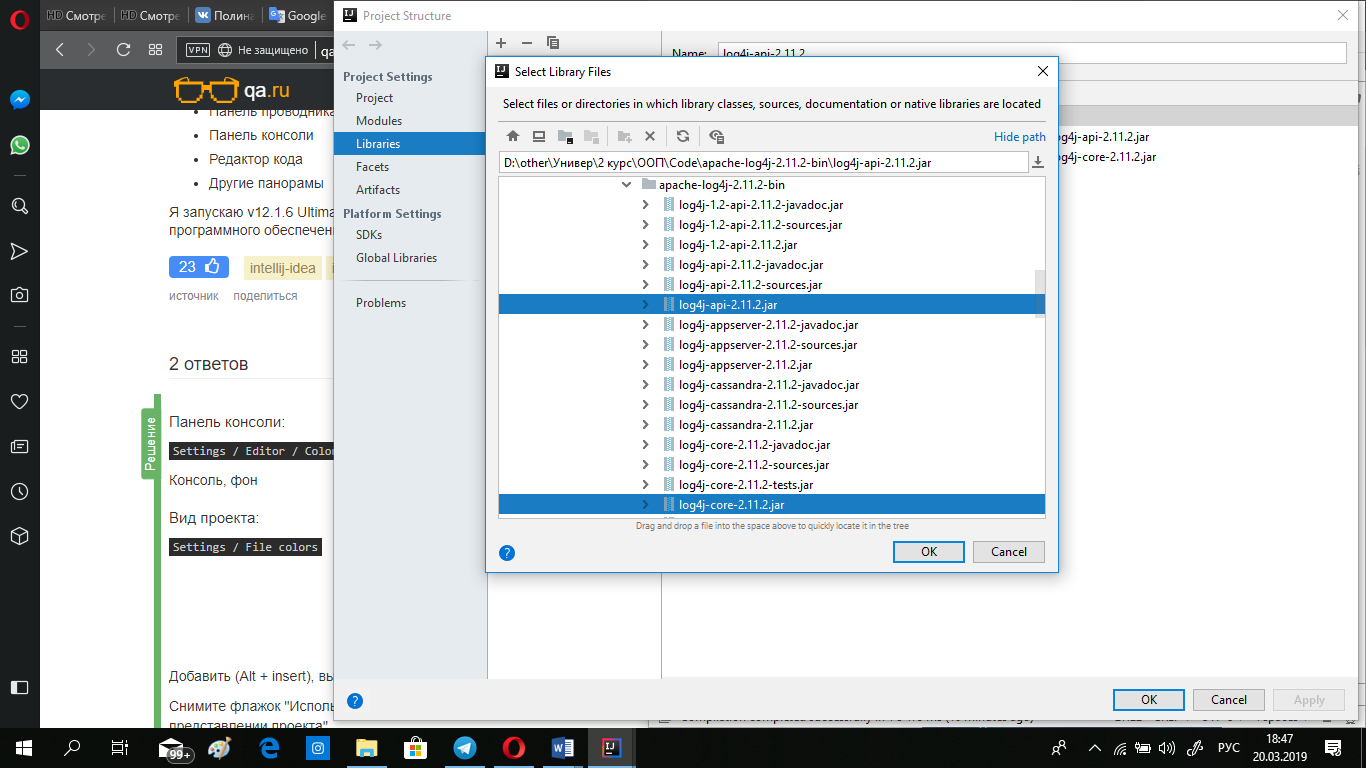
**Тема:** Фреймверк для логирования log4j.

**Цель работы:** изучение фреймверка записи логирования log4j.

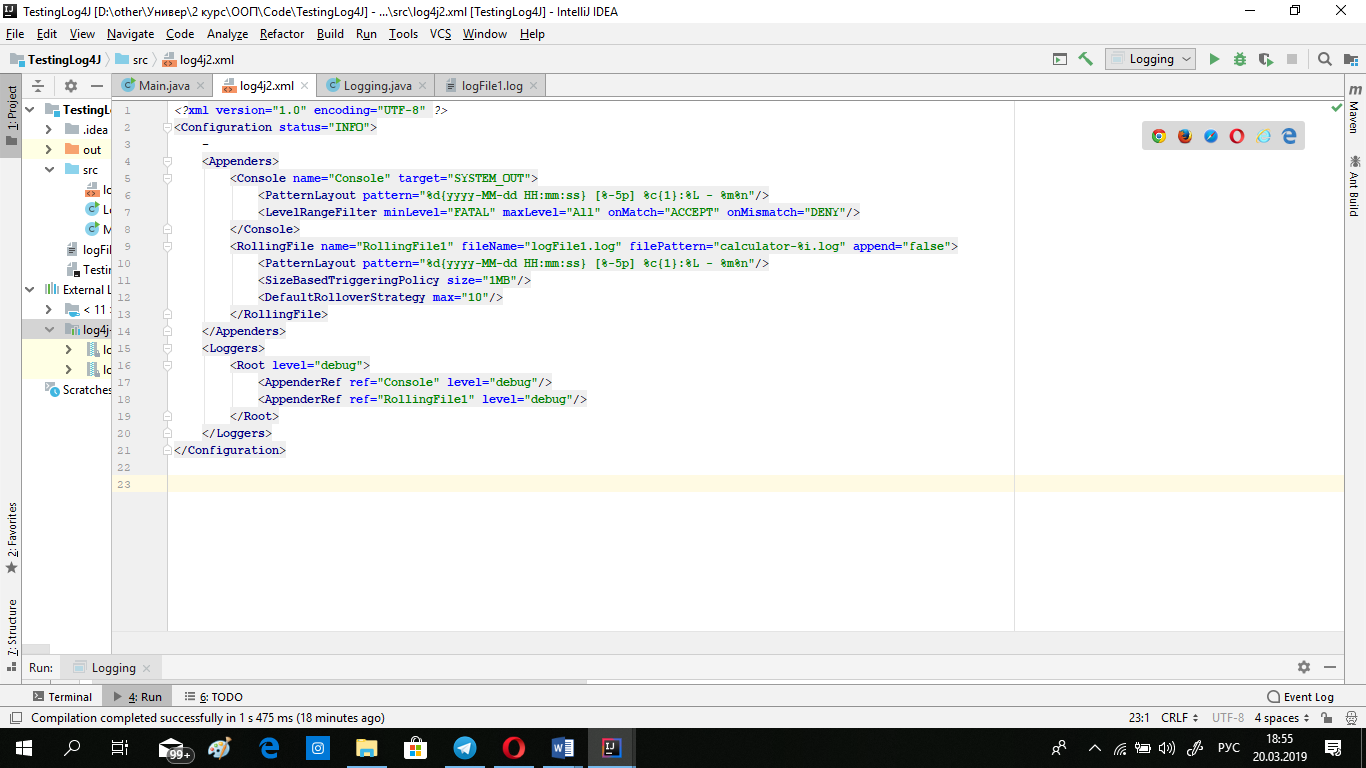
***ЗАДАНИЕ***

1. ***Подключить log4J к проекту Java.***

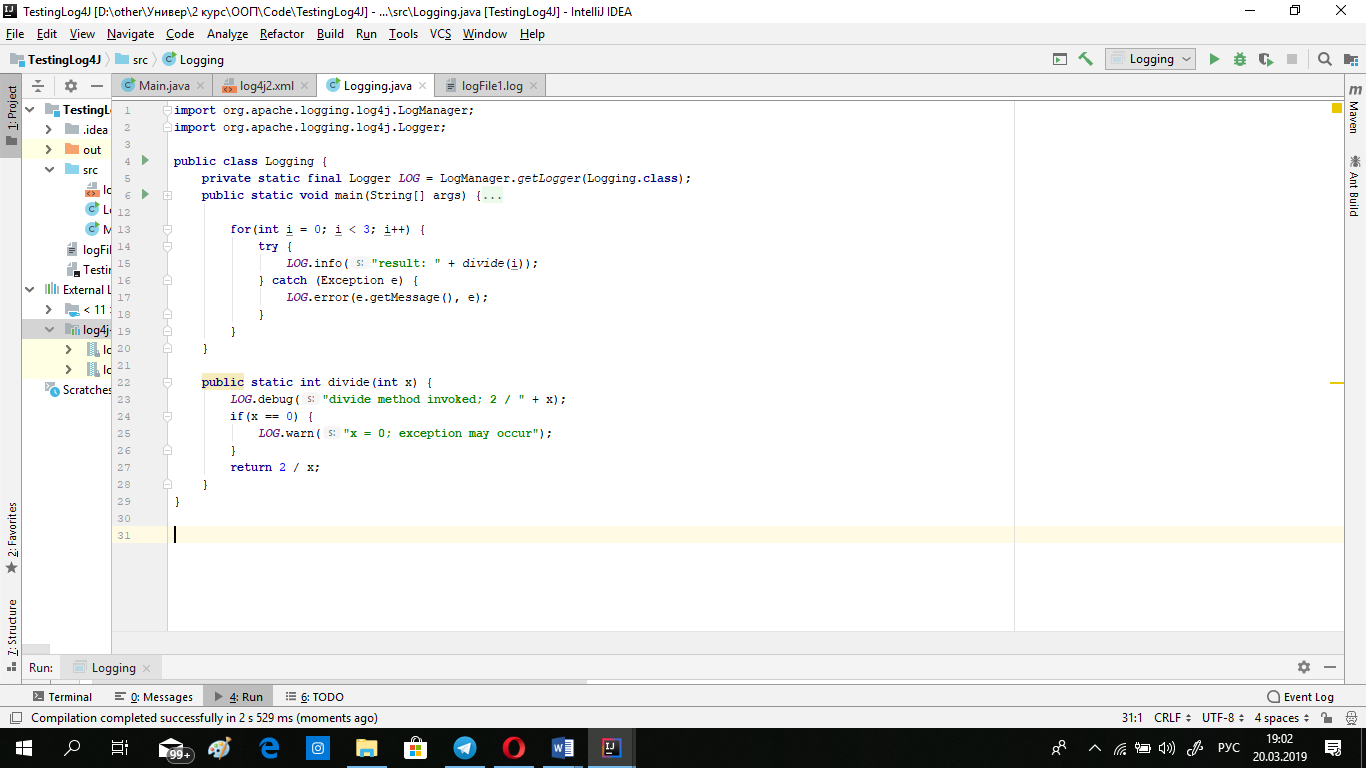
Чтобы подключить Log4J к Java-проекту нужно: скачать библиотеку фреймворка по ссылке <https://logging.apache.org/log4j/2.x/download.html>, далее распаковать архив в удобное место; создать проект **Java File 🡪 New 🡪 Project…**,дать проекту имя; зайти в **File 🡪 Project Structure…**, перейти во вкладку **Libraries**, нажать на значок «**+**», выбрать jar-файлы, которые нас интересуют – **log4j-api-2.11.2.jar** и **log4j-core-2.11.2.jar**; нажать «Окей».

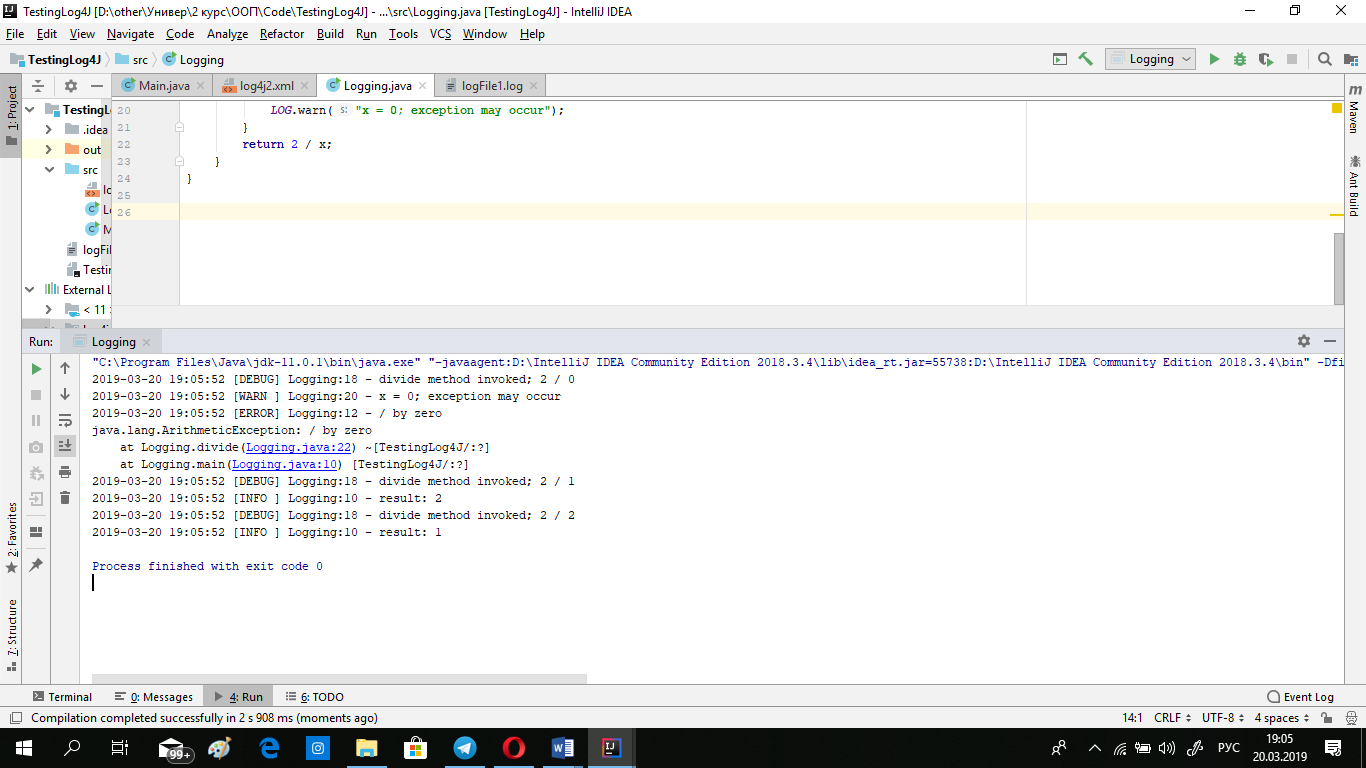


В пакете «src» создать файл “log4j2.xml”, заполнить его кодом, далее создать класс Logging, импортировать подключенные библиотеки, создать объект Logger, написать простой код и запустить выполнение.



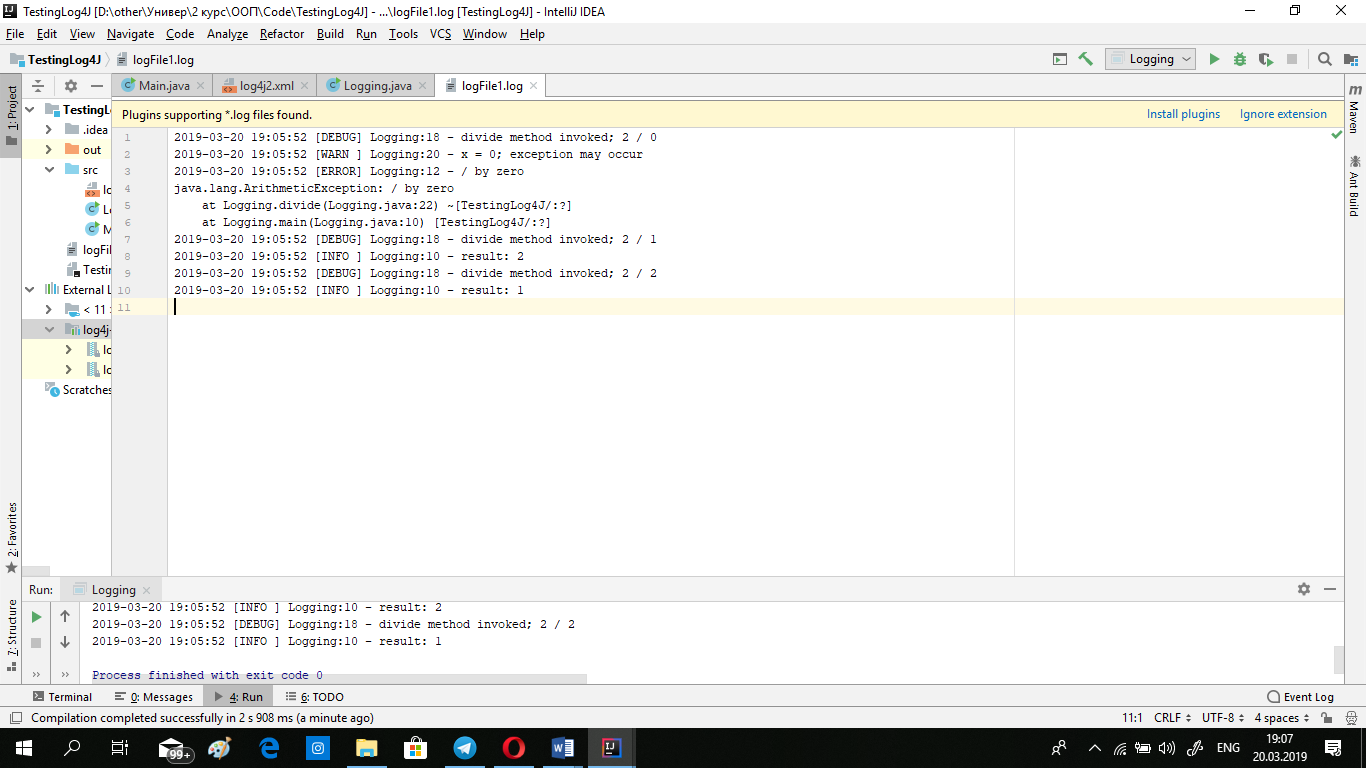
***2. Вывести сообщения уровней debug, info, warn, error, fatal в консоль.***



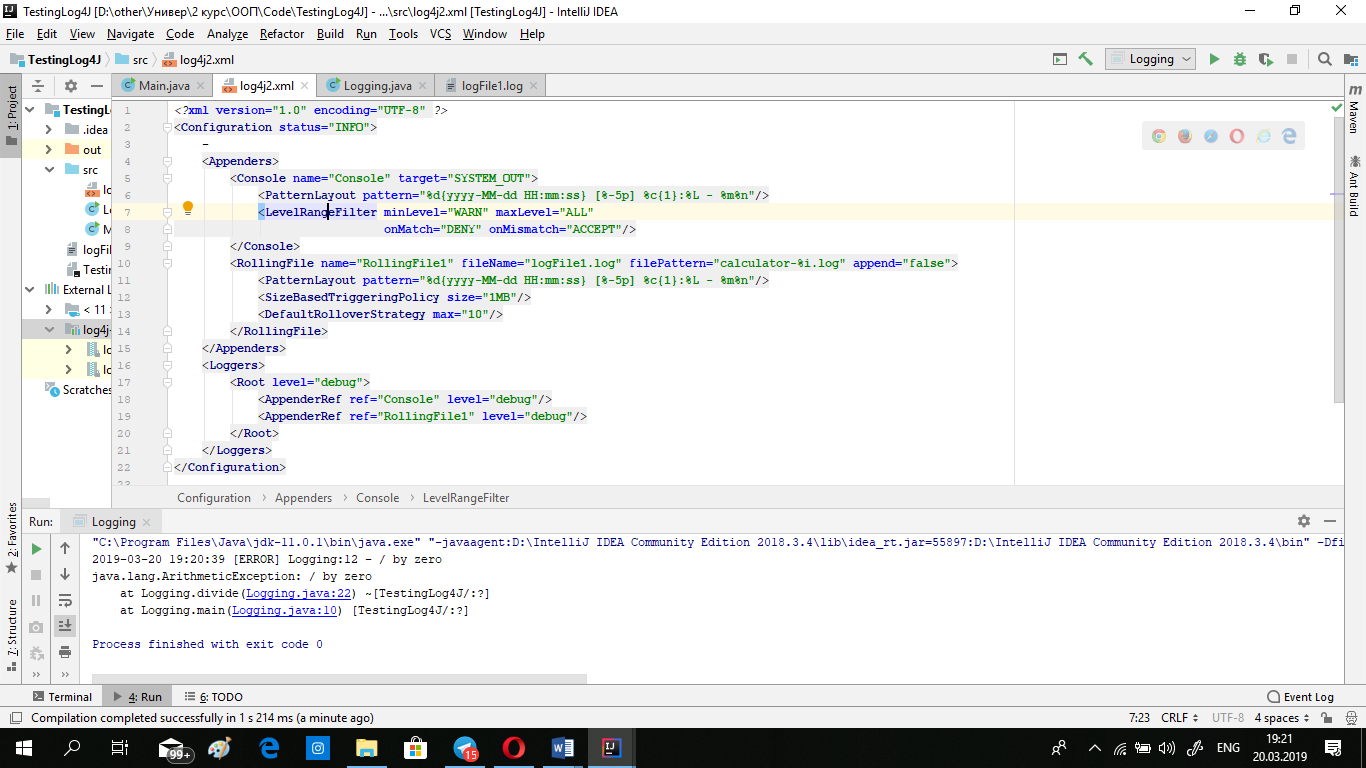
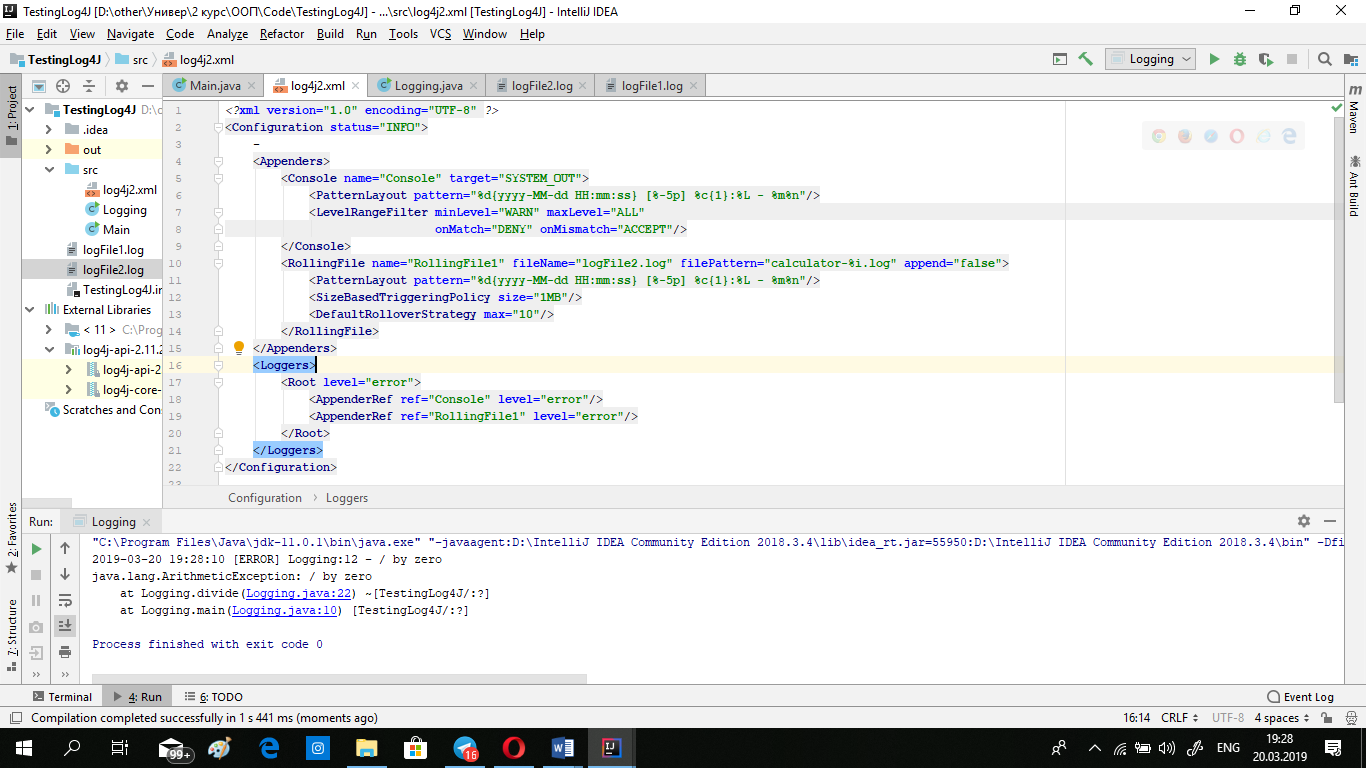


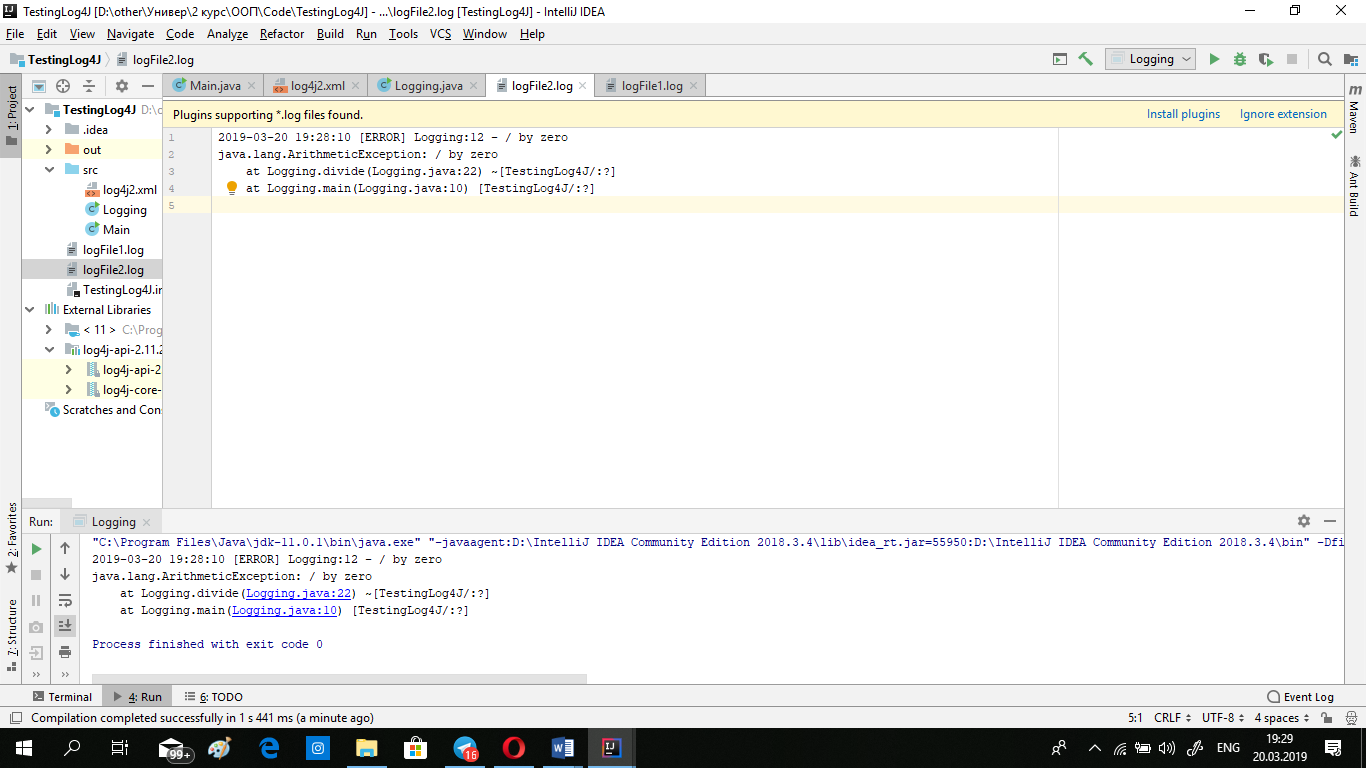
***3. Вывести сообщения уровней debug, info, warn, error, fatal в файл.***

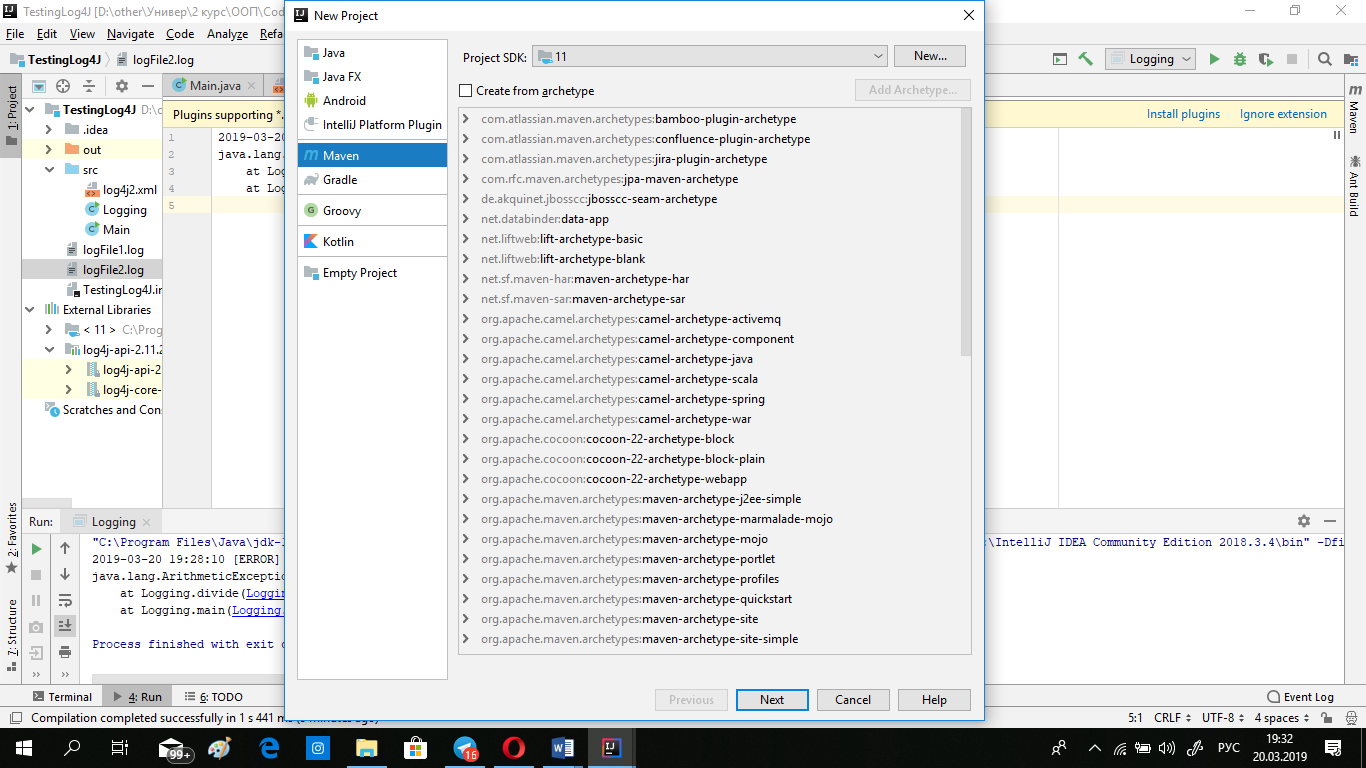
В процессе выполнения кода создается файл “logFile1.log”:



***4. Вывести сообщения уровней error, fatal в файл.***

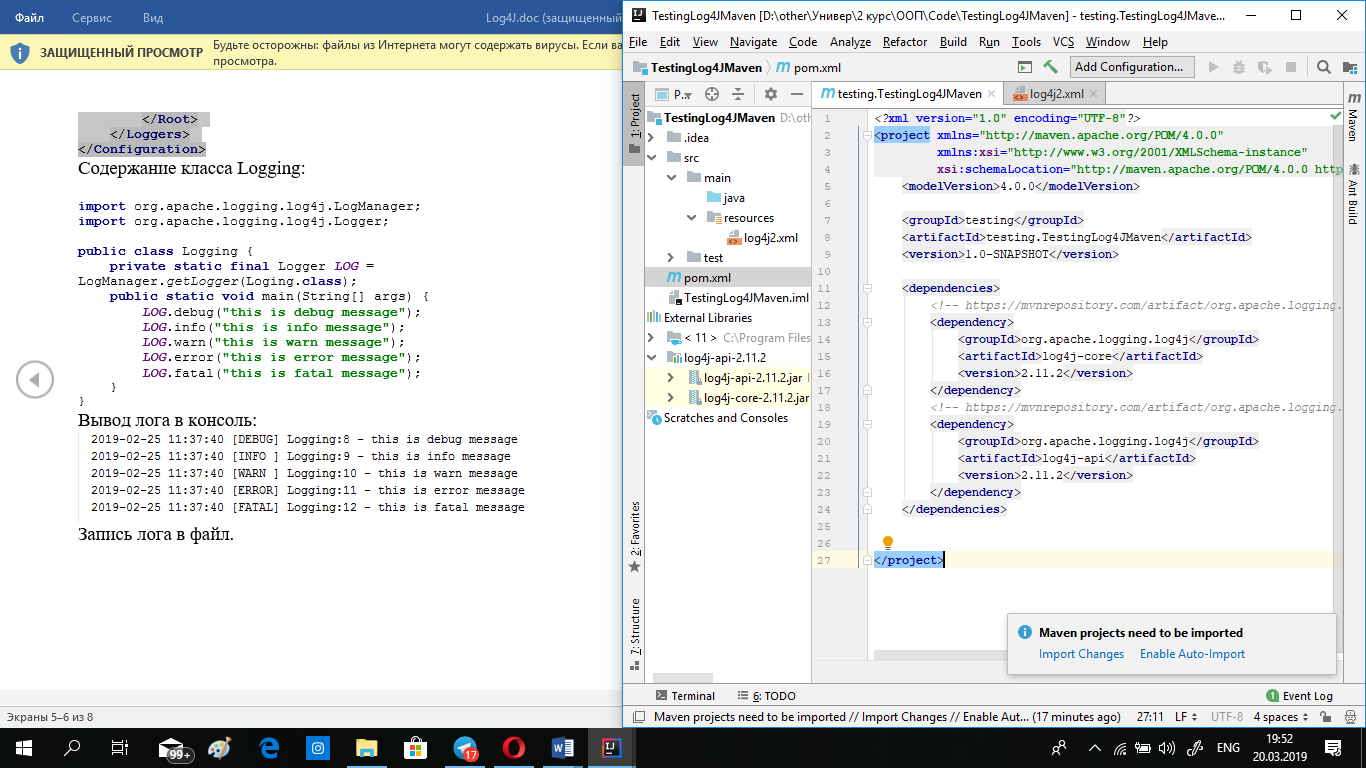
Чтобы вывести в файл сообщения уровней error, fatal, следует указать соответствующий уровень логирования в xml-файле:

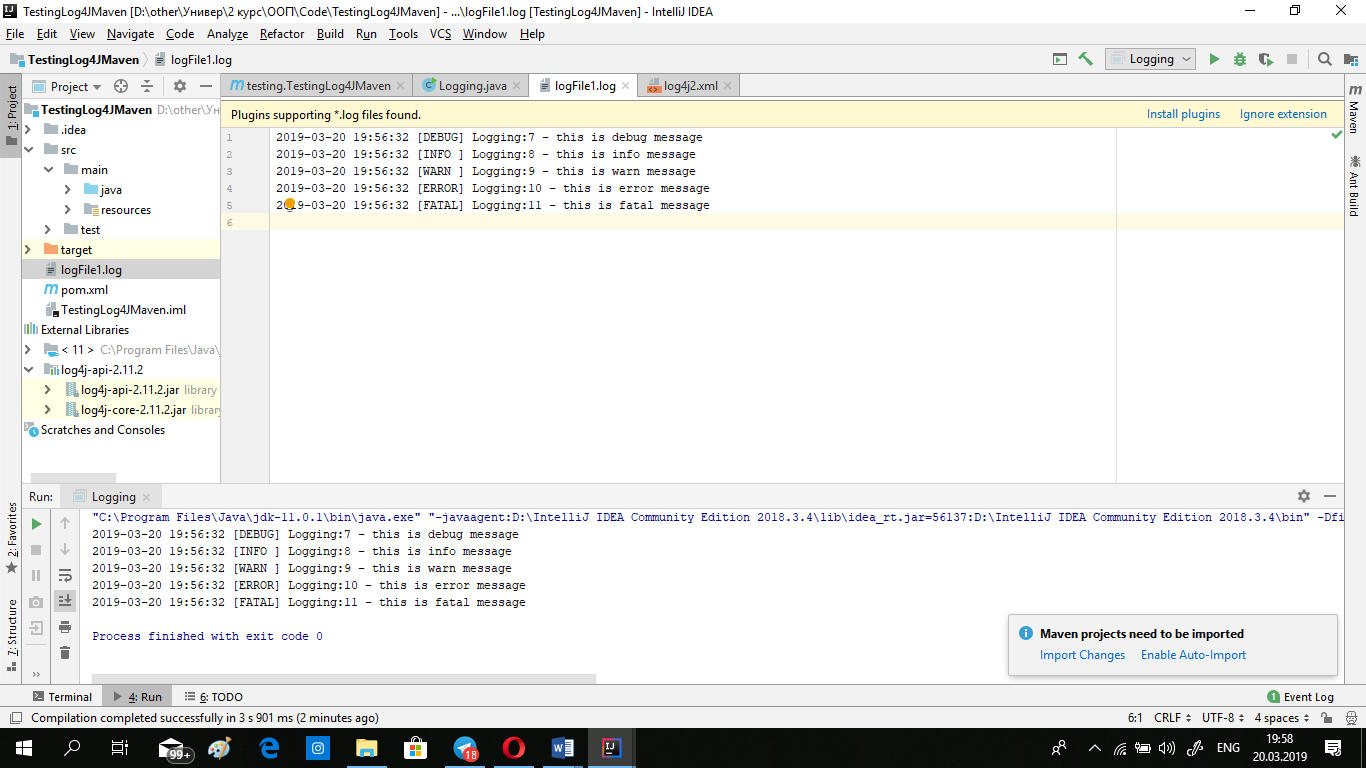


***5. Подключить log4J к проекту Maven Java.***

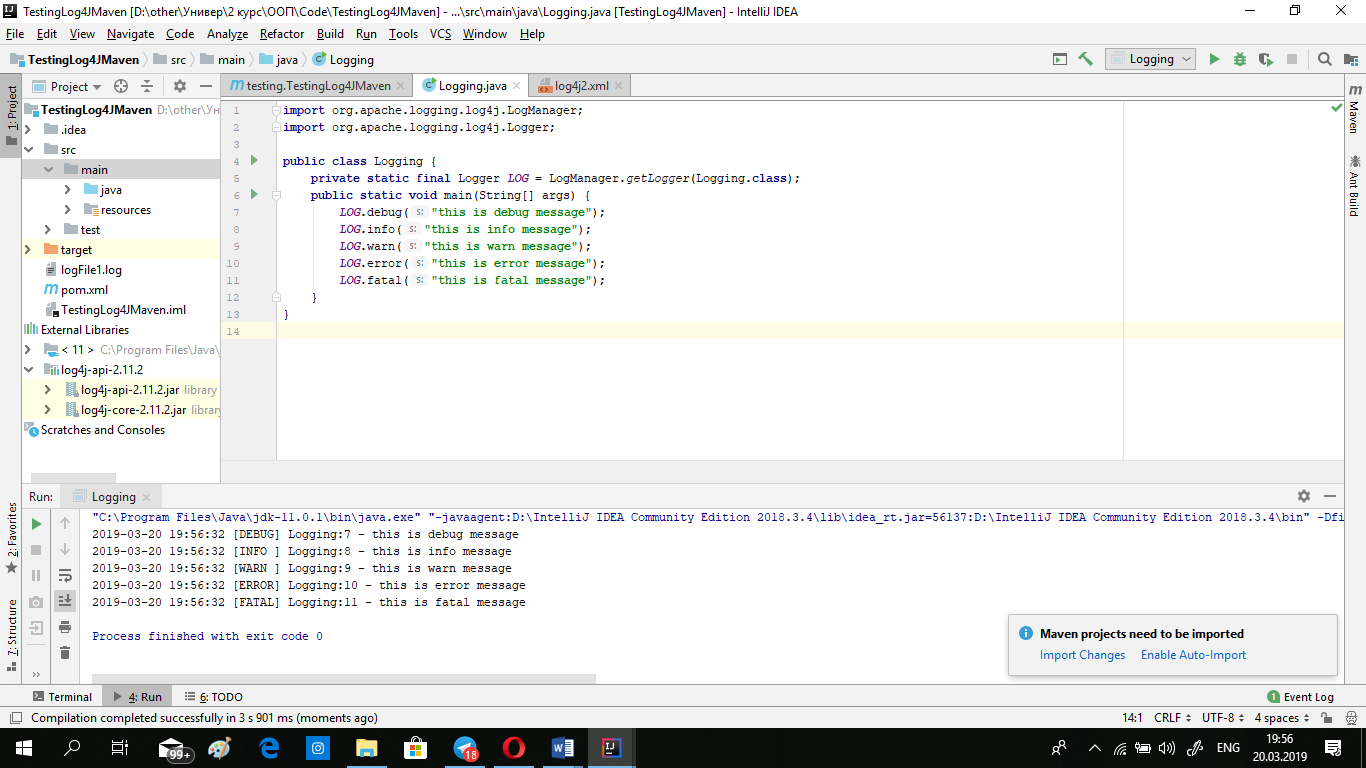
Повторяем все действия по созданию проекта, однако во всплывающей окне сразу выбираем вкладку Maven, далее даем имя проекту, соглашаемся.

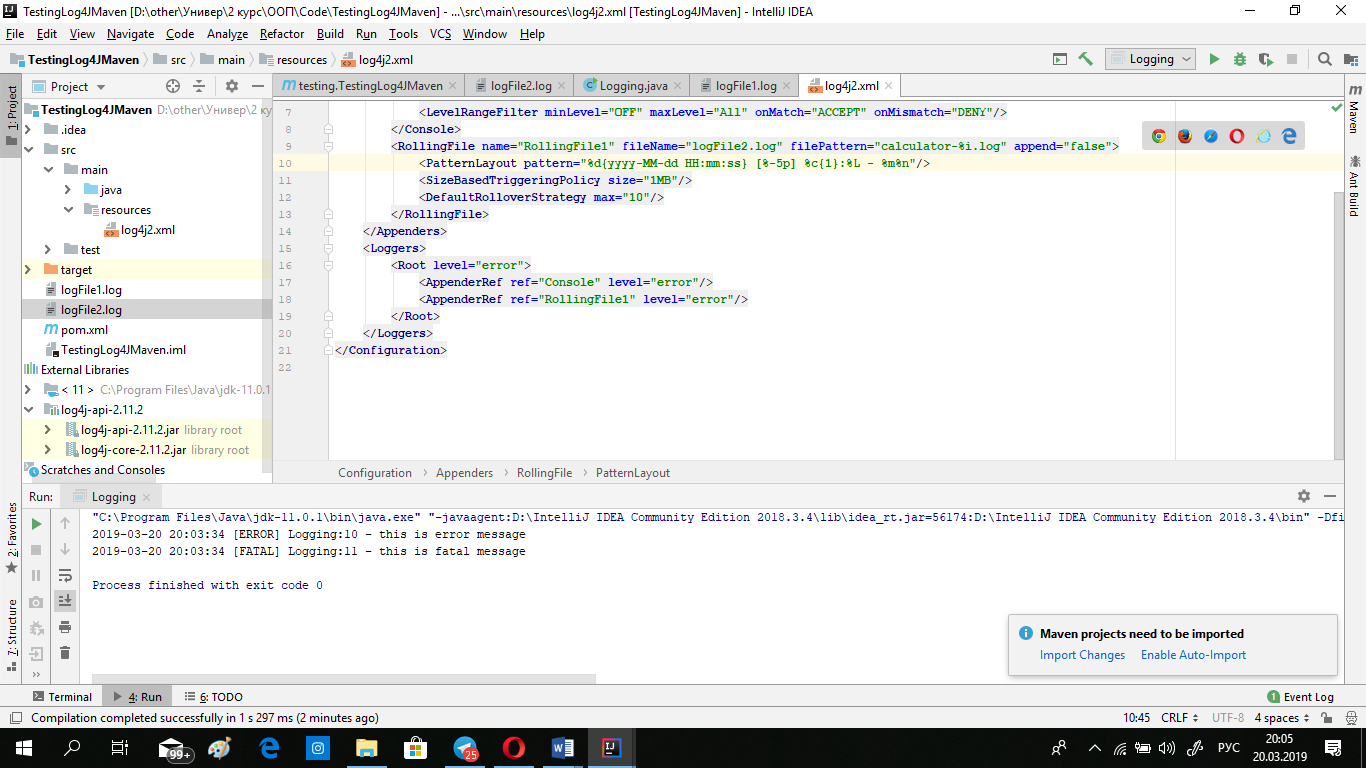
***6. Повторить пункты 2-4.***

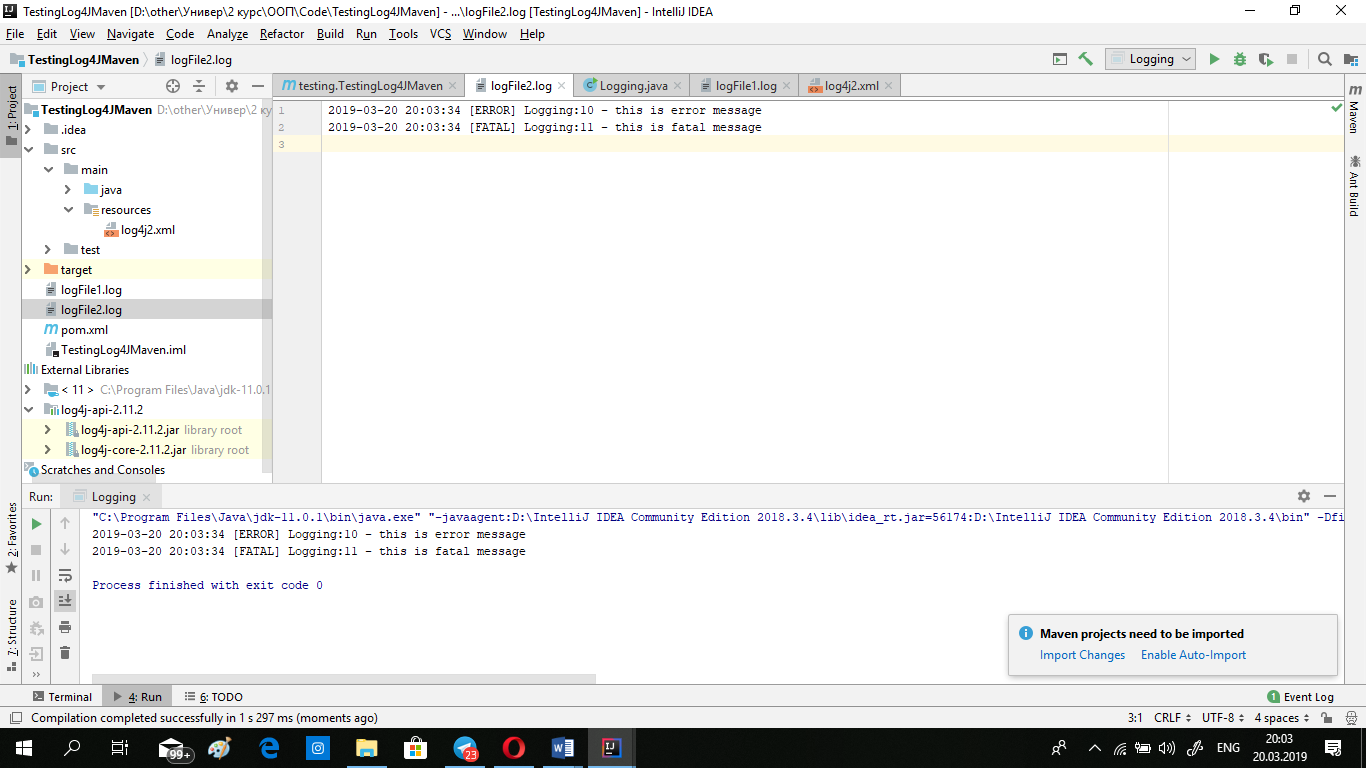
Настраиваем файл pom.xml, дописывая нужные зависимости. Создаем файл log4j.xml, как для предыдущего



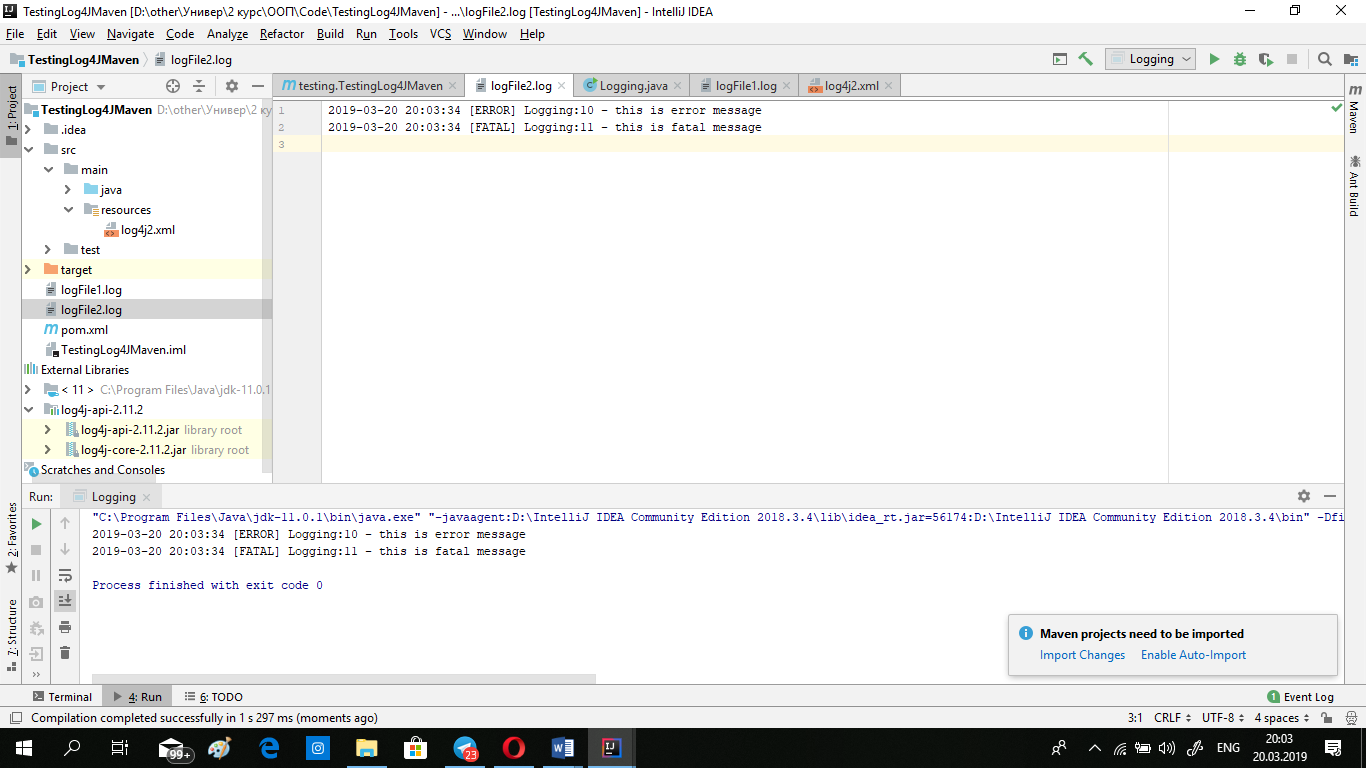
|  |  |
| --- | --- |
| Консоль | Файл “logFile1.log” |



Меняем уровень логирования на error в файле log4j2.xml, повторяем вывод на консоль и запись в файл.



🡪 🡪 🡪 Консоль



🡪 🡪 🡪 Файл “logFile2.log”

***ВЫВОД***

Выполнив лабораторную работу, я изучила фреймворк записи логирования log4j.

Узнала, что Log4J – это фреймворк для скрытия реализации рутинных операций по логированию некоторых событий, которые происходят во время работы приложения. При этом рутинные операции – ход выполнения методов, классов, блоков обработки критических ситуаций.

Изучила уровни логирования: Trace, Debug, Info, Warn, Error, Fatal.

Научилась подключать библиотеку Log4J к Maven-проекту Java и обычному проекту Java, а также записывать файлы-логи.