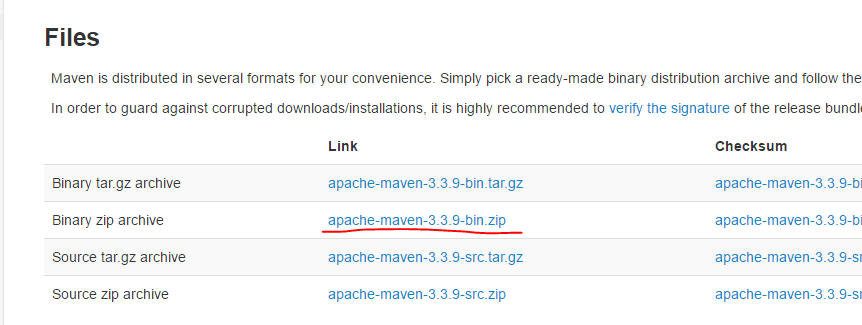
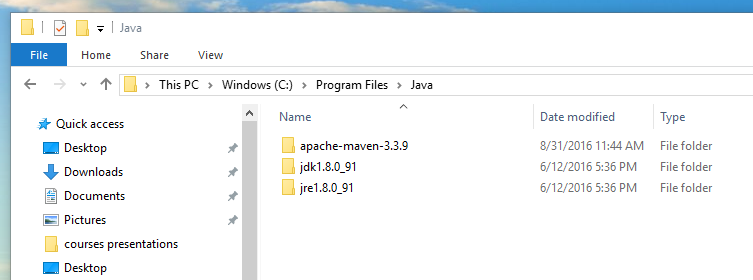
1. Установка Maven

Перейдите на сайт <https://maven.apache.org/download.cgi> и скачайте последнюю версию мейвена. Вам нужен “Binary zip archive”:



После скачивания распакуйте его любым архиватором и скопируйте содержимое на свой диск. Я распаковал его в папку “Java”, но он может находиться в любом месте:



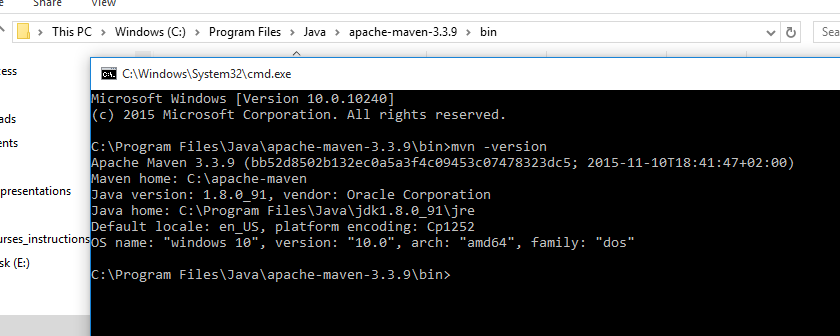
Программа “mvn”, которую мы будем использовать в дальнейшем находится в папке “bin”.

Мой полный путь это:

C:\Program Files\Java\apache-maven-3.3.9\bin

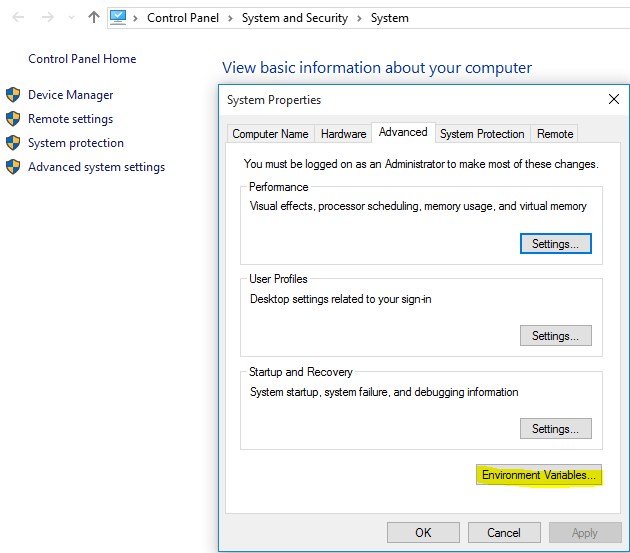
если открыть консоль из этой папки, то мы сможем запустить maven для проверки:

mvn -version

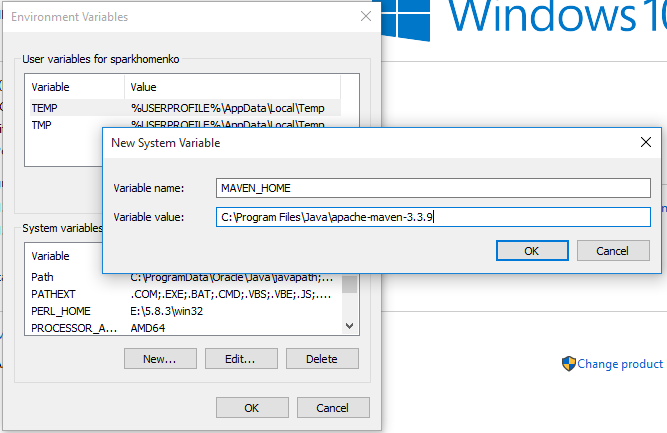


Для того, чтобы иметь возможность запускать программу из любого места в системе нужно добавить полный путь к папке “bin” в переменные окружения (environment variables).

Зайдите в «Свойства системы» (“System properties”) и во вкладке “Advanced” нажмите кнопку “Environment Variables”:

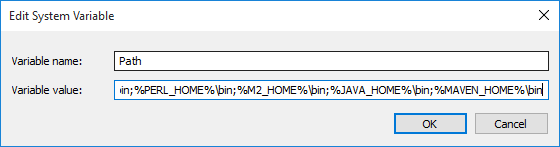


В следующем окне в разделе “System variables” добавьте переменную MAVEN\_HOME:

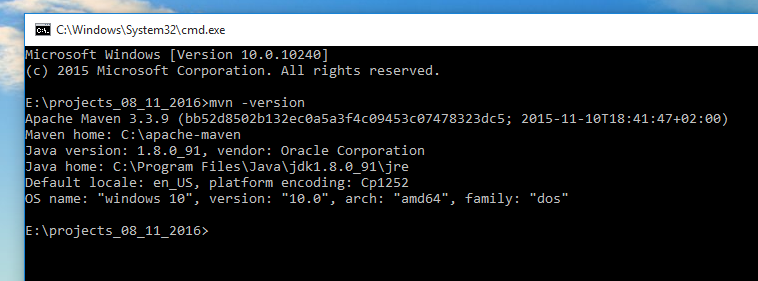


Также в переменную Path добавьте значение для запуска программы mvn из любого места на диске:  
%MAVEN\_HOME%\bin

Если у вас уже есть другие переменные, вы должны отделить последнюю с помощью точки с запятой:



Теперь откройте консоль в любом месте на диске и проверьте, что всё установлено правильно, у вас должна показываться версия мейвена и Java:



1. Создание проекта Maven

Создайте какую-нибудь папку у себя на диске, в которой будут находиться файлы и папки вашего проекта.

После этого необходимо создать еще несколько папок. Для того, чтобы программа мейвен видела наш код на джава, классы должны лежать внутри структуры папок:

src/main/java

Создайте эти папки, а также файл HelloMaven.java со следующим кодом:

class HelloMaven {

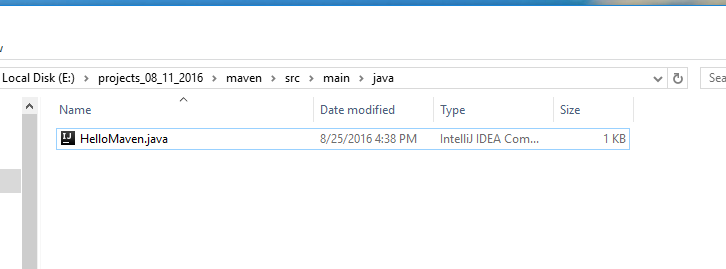
public static void main(String[] args) {

System.out.println("HelloMaven");

}

}

У меня на диске это выглядит следующим образом:



Теперь необходимо создать файл pom.xml, в корневой папке нашего проекта. Этот файл будет содержать информацию о нашем проекте и подключаемых библиотеках. Добавьте в него следующие строки:

<project>

<groupId>org.hillel</groupId>

<artifactId>mavenHelloWorld</artifactId>

<version>1.0</version>

<modelVersion>4.0.0</modelVersion>

<packaging>jar</packaging>

</project>

groupId – содержит название компании, которая поставляет этот код (вы можете написать здесь свое имя или фамилию), оно служит для обозначения уникальности вашего проекта мейвен

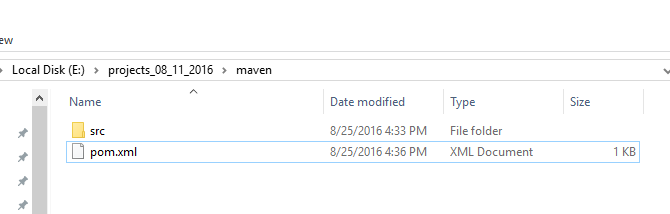
artifactId – название вашего проекта мейвен

version – текущая версия вашего проекта

modelVersion – версия для разметки файла pom.xml, для текущей версии мейвена можно использовать 4.0.0 версию (оставьте этот параметр без изменений)

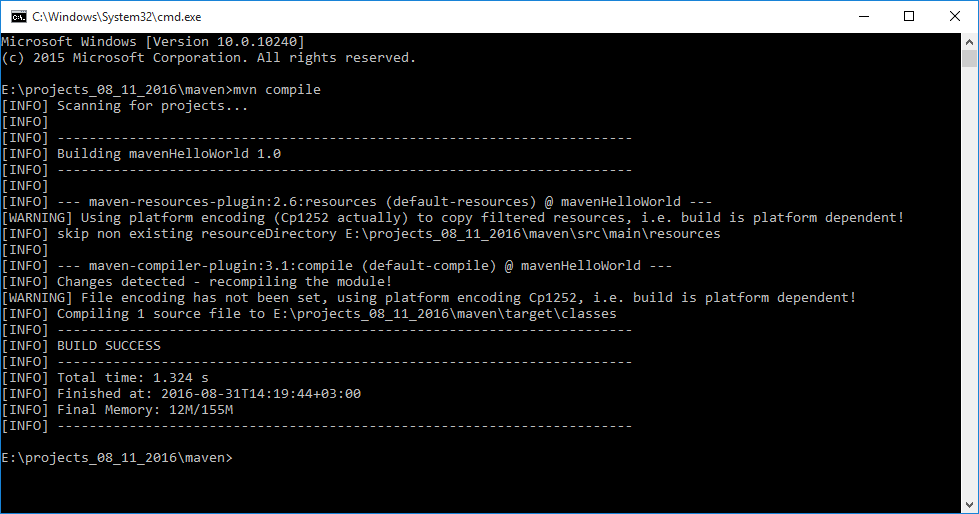
packaging – тип вашего проекта (в случае jar – будет создан jar-архив с компилированными файлами вашего проекта)

Корень вашего проекта должен теперь выглядеть следующим образом:

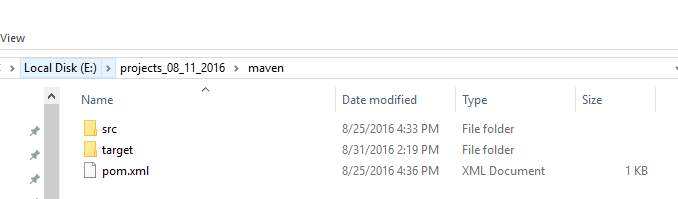


Откройте консоль в папке с файлом pom.xml и скомпилируйте ваш проект с помощью команды:

mvn compile



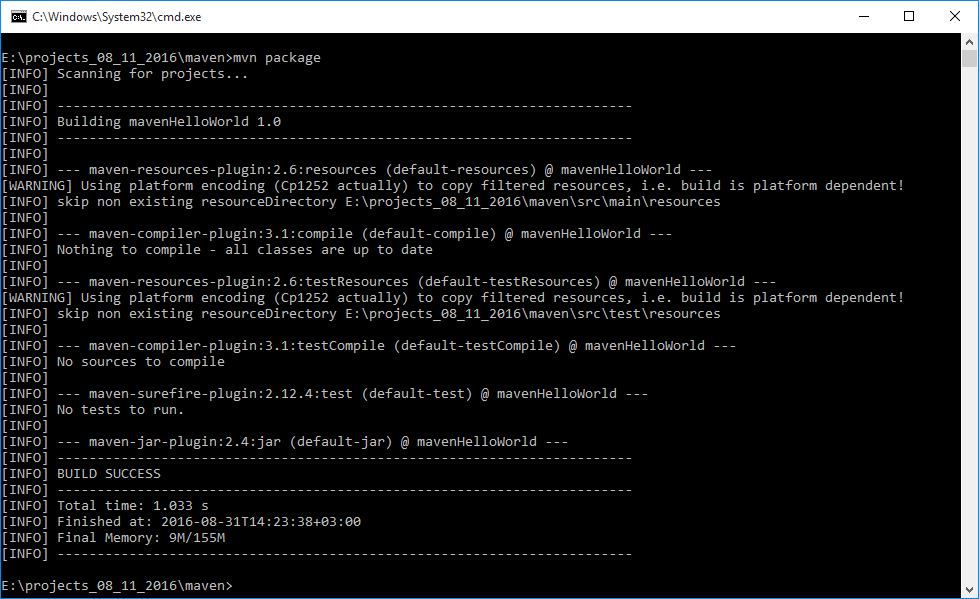
Если всё прошло хорошо, то вы должны увидеть сообщение “BUILD SUCCESS”, а в корне вашего проекта должна создаться папка “target”:



Внутри этой папке вы можете увидеть скомпилированные классы вашей программы (файлы с расширением .class)

Выполните теперь команду:

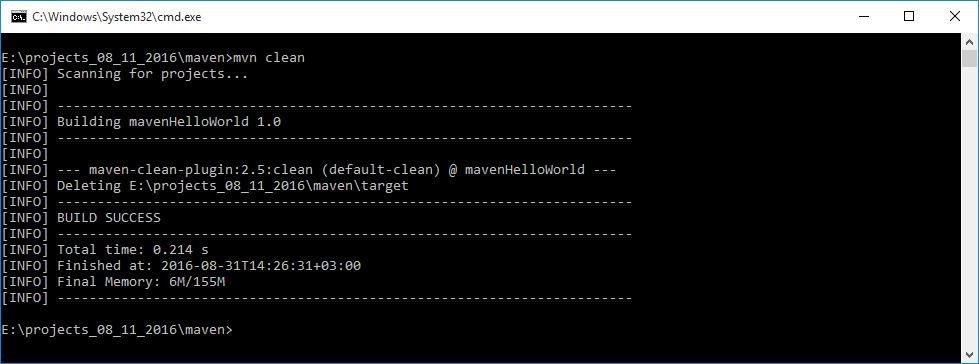
mvn package



Если всё прошло хорошо, вы должны увидеть сообщение “BUILD SUCCESS”, а в папке “target” должен появиться файл с расширением “.jar” – архив вашего проекта

Если вы хотите очистить папку “target”, вы можете воспользоваться командой:

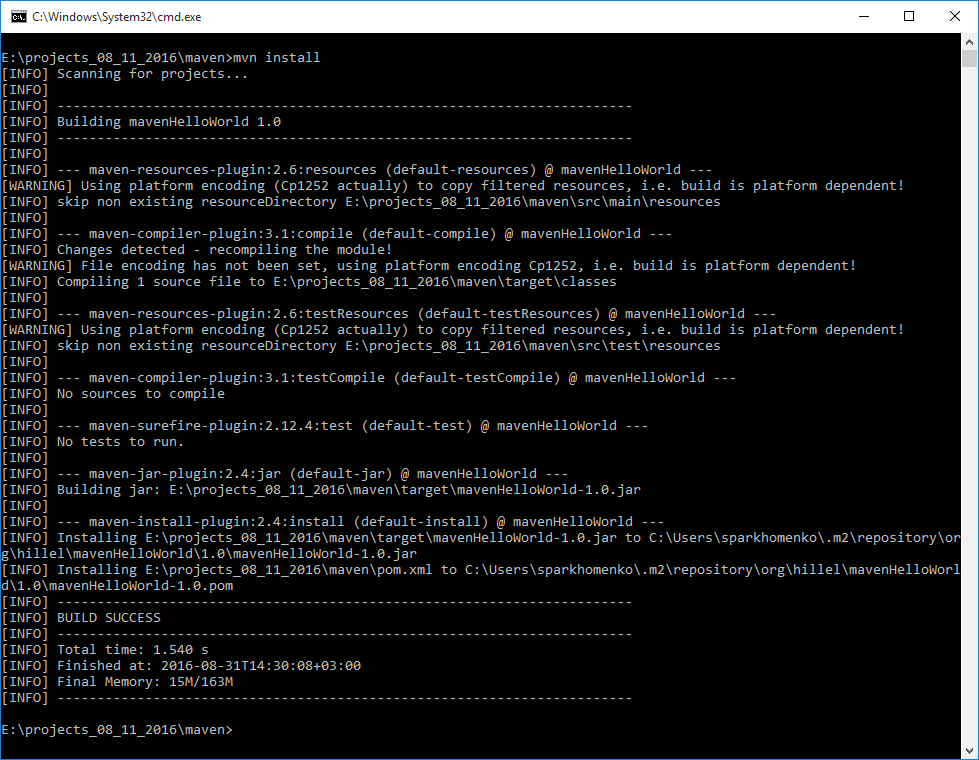
mvn clean



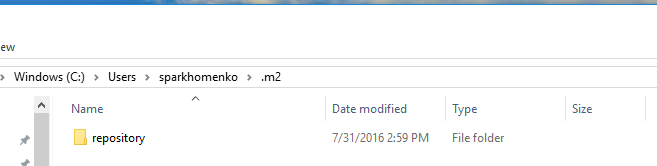
Последняя команда, которую мы рассмотрим это:

mvn install

Это команда запакует ваши классы в jar для текущего проекта, а также установит его в ваш локальный репозиторий, это нужно для того, если вы захотите импортировать его в другом мейвен проекте с помощью средств мейвен

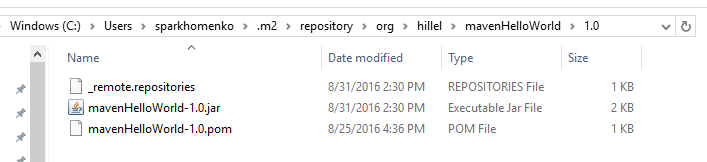


Локальный репозиторий находится в папке “.m2” в папке с текущим пользователем системы:



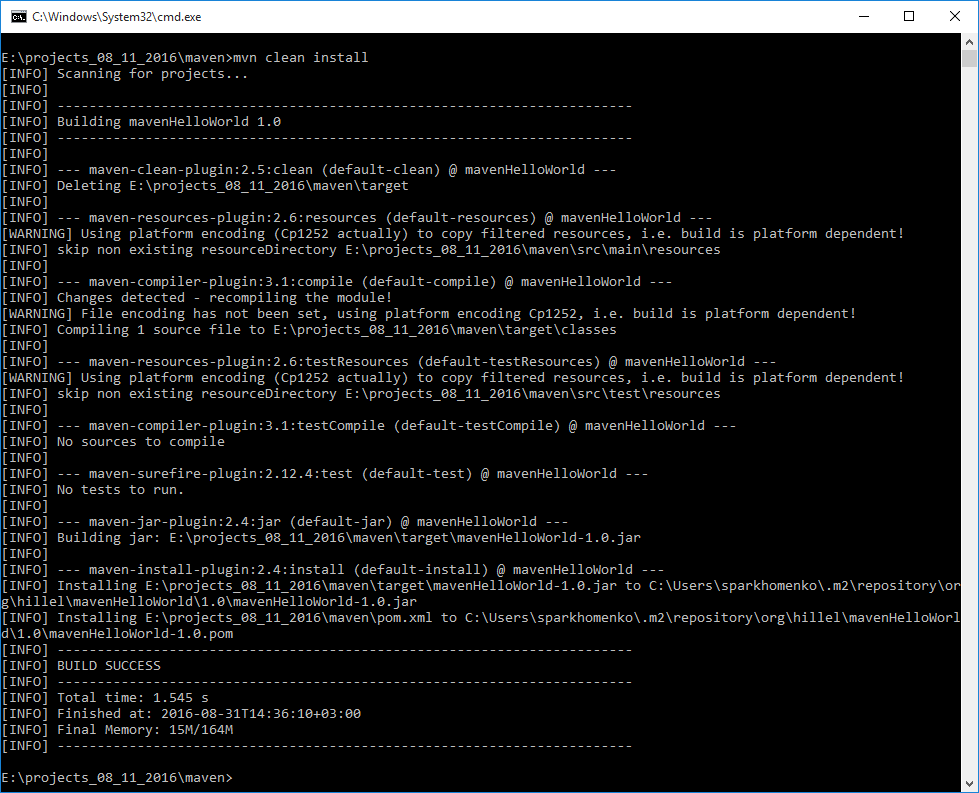
Сама библиотека внутри репозитория находится по следующему пути:

{корень репозитория}\{groupId}\{artifactId}\{version}



Обычно перед тем как создавать архив комбинируют 2 команды (вначале очистка, а потом установка):

mvn clean install

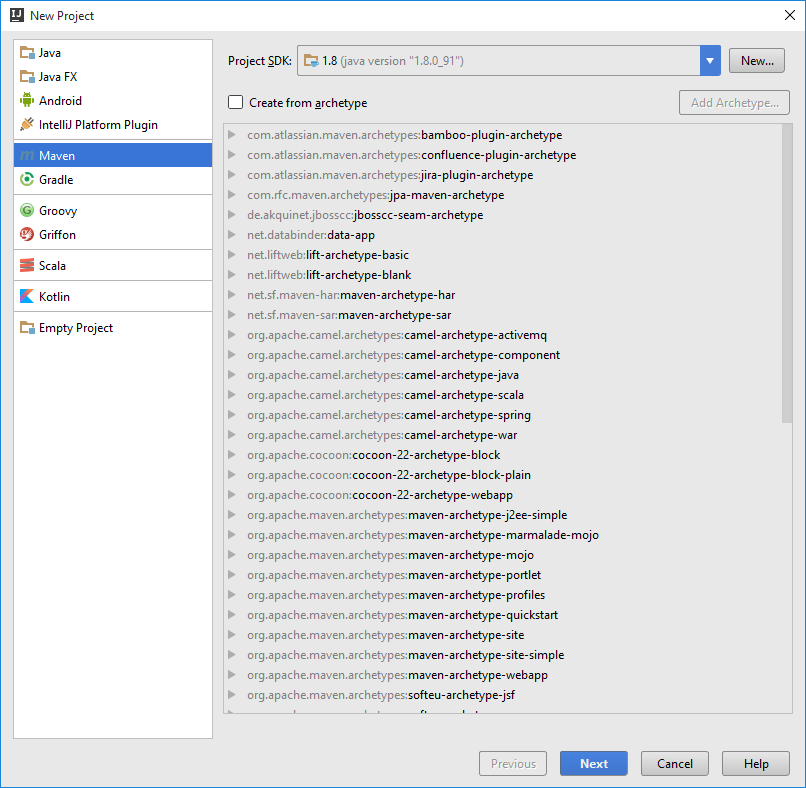


1. Создание мейвен проекта в IntelliJ IDE

Откройте IDE и создайте там новый проект:

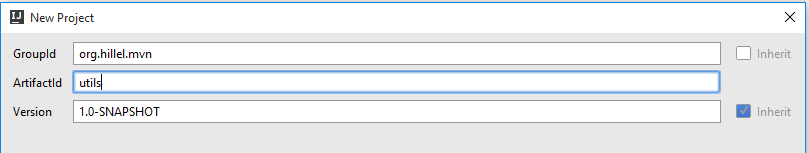
File – New – Project

В появившемся окне выберите из списка слева “Maven”:



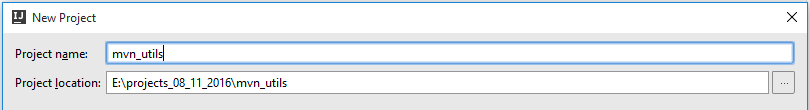
Нажмите кнопку “Next”, не выбирая больше ничего в этом окне

В следующем окне заполните обязательные поля для вашего проекта:



Нажмите “Next”

После этого укажите папку, в которой будет находиться ваш проект и нажмите кнопку “Finish”:



Структура папок в созданном проекте должна выглядеть как на рисунке:



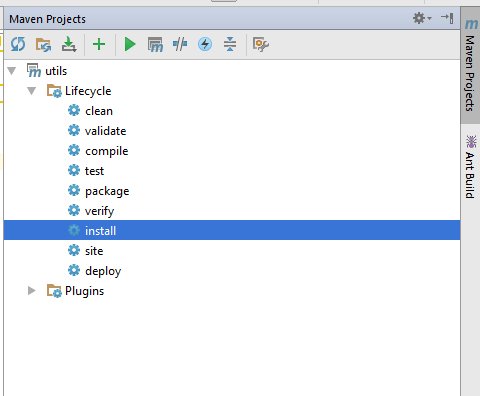
Создайте пакет и класс Utils.java внутри папки “java”.

Он будет содержать простую функцию, которая возвращает сумму двух чисел:

**public class** Utils {  
  
 **public static int** calculateSum(**int** operand1, **int** operand2) {  
 **int** result = operand1 + operand2;  
 **return** result;  
 }  
}

Вы можете использовать панель справа для выполнения maven-команд. Это всё те же команды, которые мы вводили в консоли, только теперь они встроены в IDE для более удобного вызова.

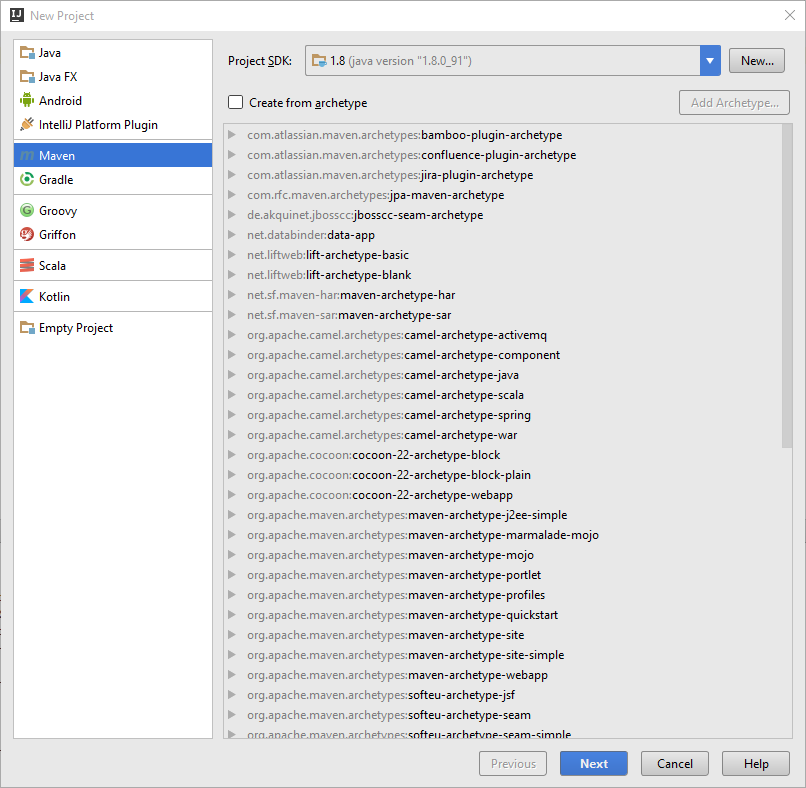
Щелкните на команду “install” и нажмите зеленый треугольник вверху этого окна для ее запуска:



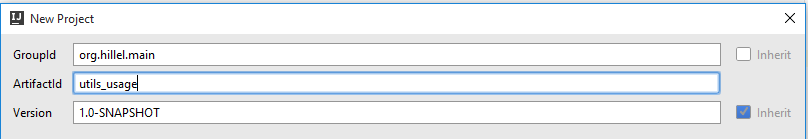
Если всё прошло хорошо, в папке “target” должен был создаться файл-архив вашего проекта, также этот файл должен был установиться в ваш локальный репозиторий.

1. Использование созданной библиотеки в другом проекте Maven

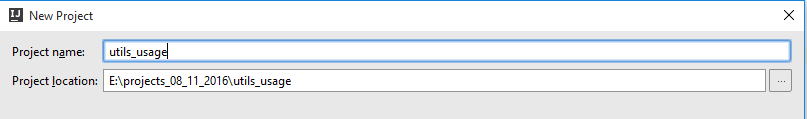
Создайте новый проект в IntelliJ Idea. Выберите, что это будет maven-проект, также как вы делали для предыдущего:



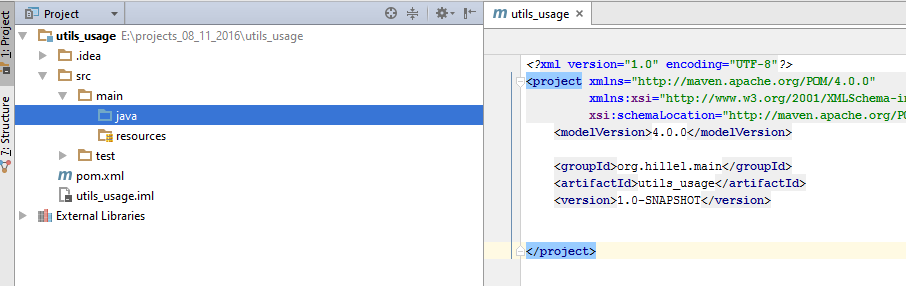
Укажите необходимые параметры мейвен-проекта:



Теперь папку, в которую будут скопированы файлы проекта:



Нажмите кнопку “Finish”



Создайте новый класс Calculator.java внутри папки java со следующим содержимым:

**public class** Calculator {  
  
 **public static void** main(String[] args) {  
 **int** sum = Utils.calculateSum(3, 5);  
 }  
}

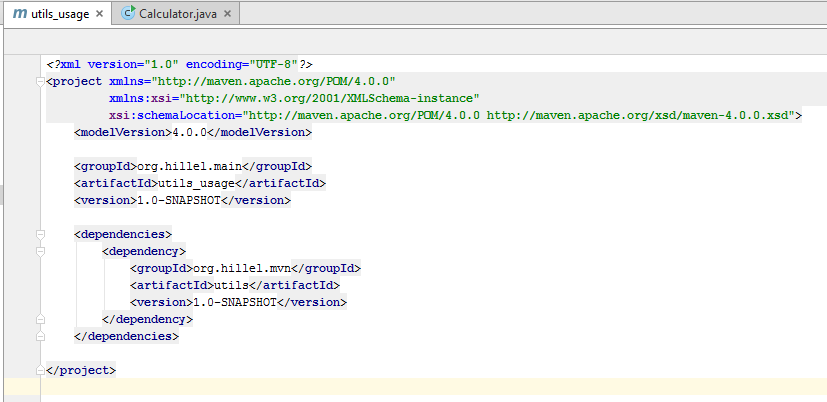
IDE показывает сейчас ошибку компилляции, так как наш проект ничего не знает о классе Utils и его методе “calculateSum”. Чтобы использовать этот метод мы должны импортировать архив jar с этим классом.

В мейвен-проекте достаточно указать зависимость (название проекта, в котором находится нужный джар), и он автоматически загрузится из репозитория, в который мы его установили с помощью команды мейвена “install”.

Добавьте следующие строки в файл pom.xml текущего проекта:

<**dependencies**>  
 <**dependency**>  
 <**groupId**>org.hillel.mvn</**groupId**>  
 <**artifactId**>utils</**artifactId**>  
 <**version**>1.0-SNAPSHOT</**version**>  
 </**dependency**>  
</**dependencies**>

Весь файл pom.xml будет выглядеть следующим образом:



Если у вас не включено автоматическое обновление изменений для мейвена щелкните правой кнопкой внутри файла pom.xml и в появившемся меню выберите:

Maven – Reimport

Теперь, если вы запустите функцию main проекта, наша созданная вспомогательная функция из проекта “utils” должна работать правильно.