

What you need to do for project 2 (windows):

1. Install
 - Jdk
 - Sbt
 - Apache spark
 - hadoop.dll
 - winutils.exe
 - Install vc redist x64
2. Setup
3. Edit Scala files
4. Create a jar file
5. Test your code.

Apache spark 2.4.7

Java 8

Install

JDK 8:

uto-em-announcements - Aug 2nd



PkgBot APP 8:17 PM

Name: Amazon Corretto JDK 8

Version: 1.8.0_302

Package Name: Amazon Corretto JDK 8-1.8.0_302.pkg



Dev: 2021-08-02 20:17:26.372254-07:00 Uploaded by: @PkgBot

Prod: 2021-08-11 05:24:05.389085-07:00 Approved by: @zthomps3

```
Select Command Prompt
Microsoft Windows [Version 10.0.19043.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Users\sghayekh>java -version
openjdk version "1.8.0_312"
OpenJDK Runtime Environment Corretto-8.312.07.1 (build 1.8.0_312-b07)
OpenJDK 64-Bit Server VM Corretto-8.312.07.1 (build 25.312-b07, mixed mode)

C:\Users\sghayekh>
```

Install SBT:

The image shows a Google search for "download sbt for windows". The search results include a link to "https://www.scala-sbt.org" and a link to "sbt Reference Manual — Installing sbt on Windows". Below the search results, there is a screenshot of the scala-sbt.org/download.html page. The page has a large blue header with the word "DOWNLOAD". Under the "Windows" section, there is a red box around the link "sbt-1.5.5.msi". Below this, there are links for "sbt-1.5.5.msi.sha256" and "sbt-1.5.5.msi.asc". Under the "Chocolatey" section, there is a code block with the command "> choco install sbt".

Google search results for "download sbt for windows".

Results include:

- <https://www.scala-sbt.org> › download
- Download - SBT**
- Download · Mac · **Windows** · Linux (deb) · Linux (rpm) · All platforms · Community Support · Commercial Support.
- <https://www.scala-sbt.org/docs/Installing-sbt-on-Windows>
- sbt Reference Manual — Installing sbt on Windows**
- Installing **sbt** on **Windows**. Install JDK ... **Download** ZIP or TGZ package and expand it.
- Windows** installer. **Download** msi installer and install it.
- <https://www.scala-sbt.org>
- sbt** - The interactive build tool

Browser address bar: scala-sbt.org/download.html

Page content:

DOWNLOAD

Windows

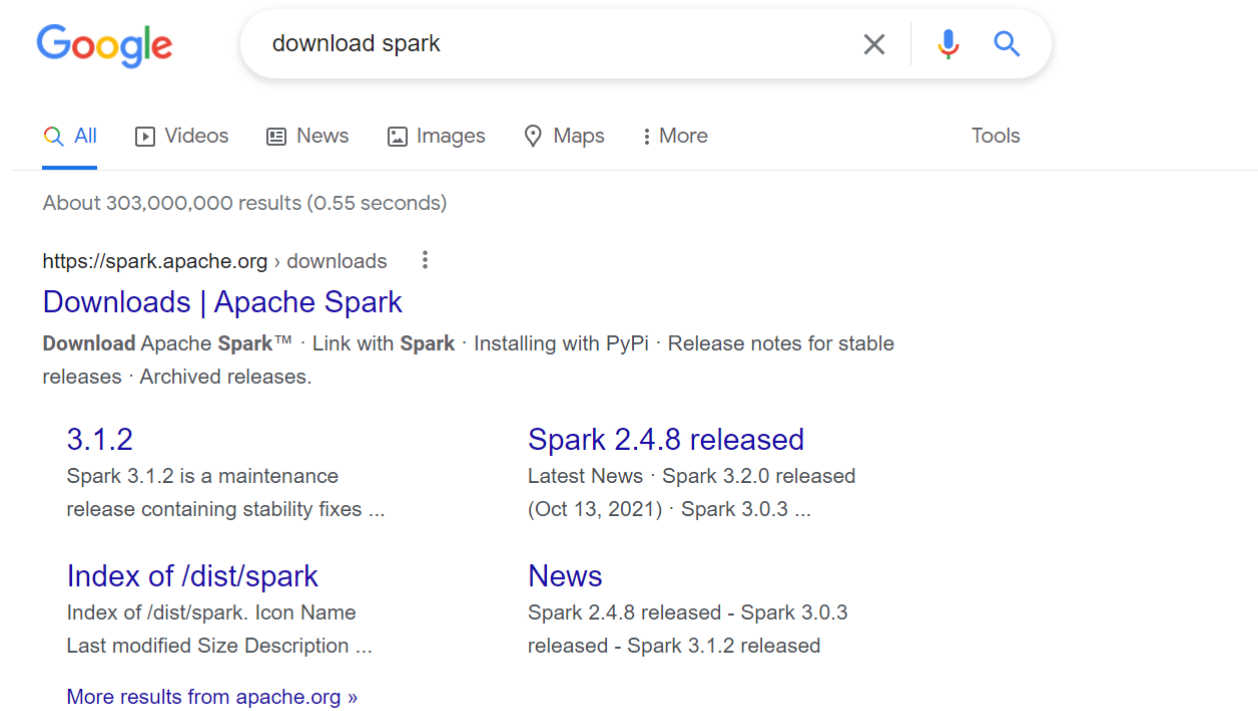
- sbt-1.5.5.msi**
- [sbt-1.5.5.msi.sha256](#)
- [sbt-1.5.5.msi.asc](#)

Chocolatey

```
> choco install sbt
```

Done!

Then install Apache spark:



The screenshot shows a Google search interface. The search bar contains the text "download spark". Below the search bar, the Google logo is on the left, and navigation links for "All", "Videos", "News", "Images", "Maps", and "More" are in the center. A "Tools" link is on the right. Below the navigation bar, it says "About 303,000,000 results (0.55 seconds)". The first search result is from "https://spark.apache.org" and is titled "Downloads | Apache Spark". Below the title, there is a link to "Download Apache Spark™" and other links like "Link with Spark", "Installing with PyPi", "Release notes for stable releases", and "Archived releases". There are two more search results visible: "3.1.2" and "Spark 2.4.8 released".

Google

download spark

All Videos News Images Maps More Tools

About 303,000,000 results (0.55 seconds)

https://spark.apache.org › downloads

Downloads | Apache Spark

Download Apache Spark™ · Link with Spark · Installing with PyPi · Release notes for stable releases · Archived releases.

3.1.2

Spark 3.1.2 is a maintenance release containing stability fixes ...

Spark 2.4.8 released

Latest News · Spark 3.2.0 released (Oct 13, 2021) · Spark 3.0.3 ...

Index of /dist/spark
















Index of /dist/spark. Icon Name Last modified Size Description ...

News

Spark 2.4.8 released - Spark 3.0.3 released - Spark 3.1.2 released

[More results from apache.org »](#)

Index of /dist/spark/spark-2.4.7

Name	Last modified	Size	Description
 Parent Directory		-	
 SparkR_2.4.7.tar.gz	2020-09-08 07:13	310K	
 SparkR_2.4.7.tar.gz.asc	2020-09-08 07:13	819	
 SparkR_2.4.7.tar.gz.sha512	2020-09-08 07:13	207	
 pyspark-2.4.7.tar.gz	2020-09-08 07:13	208M	
 pyspark-2.4.7.tar.gz.asc	2020-09-08 07:13	819	
 pyspark-2.4.7.tar.gz.sha512	2020-09-08 07:13	210	
 spark-2.4.7-bin-hadoop2.6.tgz	2020-09-08 07:13	221M	
 spark-2.4.7-bin-hadoop2.6.tgz.asc	2020-09-08 07:13	819	
 spark-2.4.7-bin-hadoop2.6.tgz.sha512	2020-09-08 07:13	268	
 spark-2.4.7-bin-hadoop2.7.tgz	2020-09-08 07:13	223M	
 spark-2.4.7-bin-hadoop2.7.tgz.asc	2020-09-08 07:13	819	
 spark-2.4.7-bin-hadoop2.7.tgz.sha512	2020-09-08 07:13	268	
 spark-2.4.7-bin-without-hadoop-scala-2.12.tgz	2020-09-08 07:13	140M	
 spark-2.4.7-bin-without-hadoop-scala-2.12.tgz.asc	2020-09-08 07:13	819	

1. Download
2. Unzip
3. and copy-paste the files and folders:
 - in drive C → spark-local→spark.
 - Create another folder in drive C→spark-local→hadoop→bin

From the following repository:

<https://github.com/steveloughran/winutils/tree/master/hadoop-2.7.1/bin>

Open: hadoop.dll

- Download and paste it in Hadoop→ bin
- It also needs to be paste in another folder: Root drive (C) → windows

Also

Open: winutils.exe

- Download and paste it in Hadoop→ bin

Need to Install vc redist x64

[All](#) [Books](#) [Videos](#) [News](#) [Images](#) [More](#)

About 185,000 results (0.63 seconds)

[https://www.microsoft.com › en-us › download › details](https://www.microsoft.com/en-us/download/details) ⋮

[Download Microsoft Visual C++ 2010 Service Pack 1 ...](#)

May 12, 2021 — Download **Microsoft Visual C++ 2010 Service Pack 1 Redistributable Package** MFC Security Update from Official Microsoft Download Center.

[https://www.microsoft.com › Download › confirmation](https://www.microsoft.com/Download/confirmation) ⋮

[Download Microsoft Visual C++ 2010 Service Pack 1 ...](#)

... vulnerability in MFC applications that are built with Visual Studio 2010 and ship the **Microsoft Visual C++ 2010 Service Pack 1 Redistributable Package**.

Microsoft Visual C++ 2010 Service Pack 1 Redistributable Package MFC Security Update

Important! Selecting a language below will dynamically change the complete page content to that language.

Select Language: ⌵

[Download](#)

A security issue has been identified leading to a vulnerability in MFC applications that are built with Visual Studio 2010 and ship the Microsoft Visual C++ 2010 Service Pack 1 Redistributable Package.

[⊕ Details](#)

[⊕ System Requirements](#)

Choose the download you want

<input type="checkbox"/> File Name	Size
<input type="checkbox"/> vcredist_x86.exe	8.6 MB
<input type="checkbox"/> vcredist_ia64.exe	2.9 MB
<input checked="" type="checkbox"/> vcredist_x64.exe	9.8 MB

Done!.

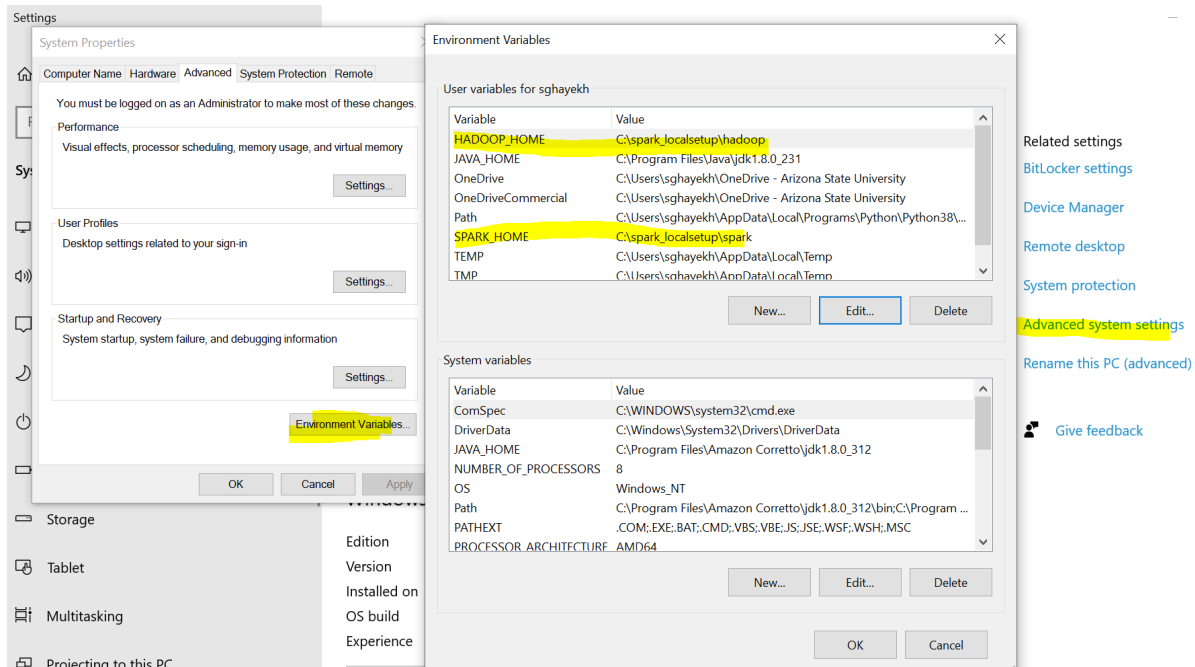
Setup

The go to the following path to change the environment variables and path:

The screenshot shows the Windows Control Panel window with the address bar set to 'Control Panel > All Control Panel Items'. Below the window, the Windows Settings app is open to the 'About' page, displaying device specifications. The specifications include:

- Device name: EN4182203W
- Full device name: EN4182203W.fulton.ad.asu.edu
- Processor: Intel(R) Core(TM) i7-8665U CPU @ 1.90GHz 2.11 GHz
- Installed RAM: 16.0 GB (15.8 GB usable)
- Device ID: 80CE9960-C1C6-40F7-AC16-FD8805E8290D

Open advance system setting→environment variables→Create two folders HADOOP_HOME and SPARK_HOME with their address in drive C.

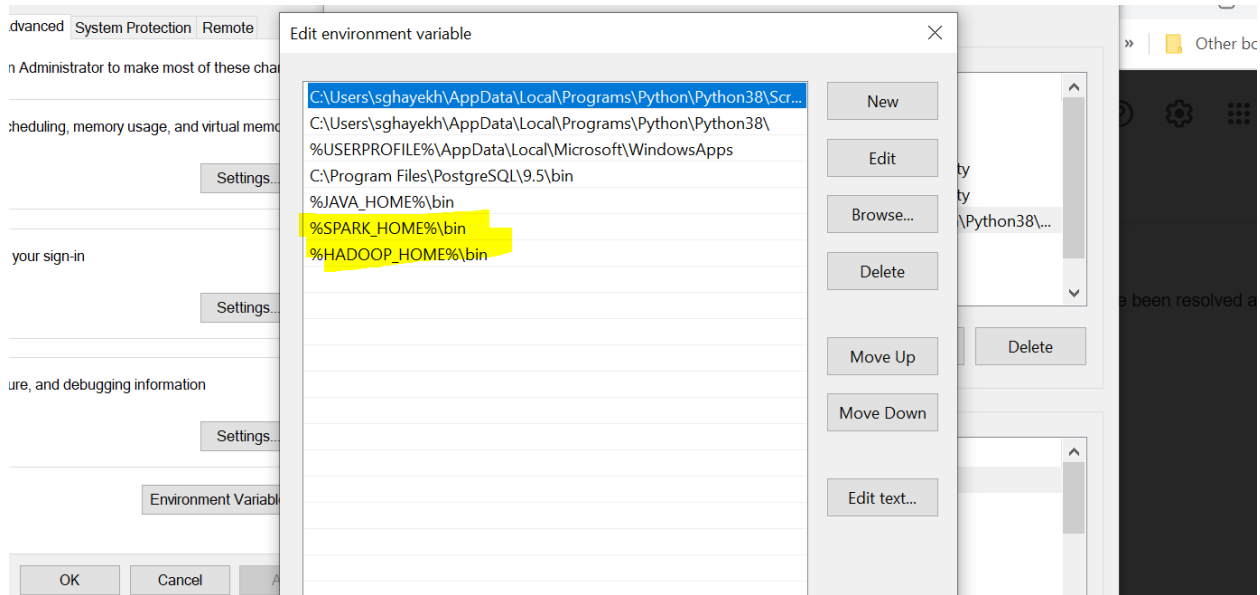


Go to path:

Create two new lines; to access to their bin folder

%SPARK_HOME%\bin

%HADOOP_HOME%\bin



RESTART YOUR COMPUTER>

After restarting:

1- Checking JAVA:

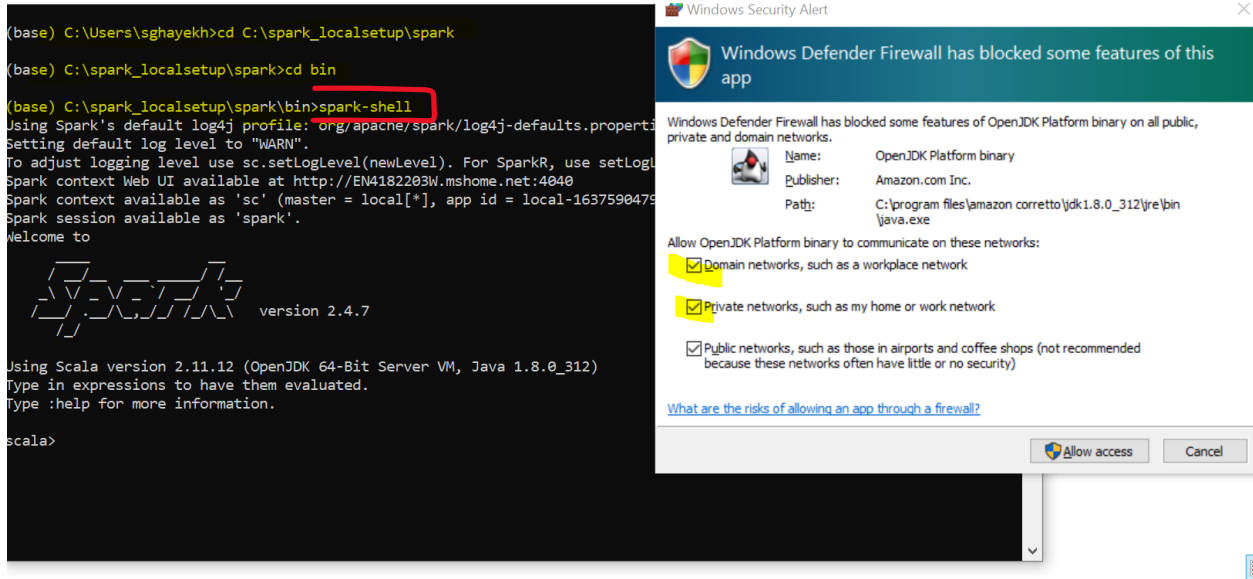
Open command line to see Java is working or not:

```
C:\Users\sghayekh>java
Usage: java [-options] class [args...]
           (to execute a class)
 or java [-options] -jar jarfile [args...]
           (to execute a jar file)
where options include:
    -d32          use a 32-bit data model if available
    -d64          use a 64-bit data model if available
    -server       to select the "server" VM
                  The default VM is server.

    -cp <class search path of directories and zip/jar files>
    -classpath <class search path of directories and zip/jar files>
                  A ; separated list of directories, JAR archives,
                  and ZIP archives to search for class files.
    -D<name>=<value>
                  set a system property
    -verbose:[class|gc|jni]
                  enable verbose output
    -version      print product version and exit
    -version:<value>
                  Warning: this feature is deprecated and will be removed
                  in a future release.
                  require the specified version to run
    -showversion  print product version and continue
```


2-Checking SPARK:

Spark-shell



Done!

Hot Zone analysis:

Edit Scala files

Download the template from coursera:

<https://www.coursera.org/learn/cse511/programming/FJqxG/project-2-hotspot-analysis-autograder-2-0>

Project 2: Hot Spot Analysis

Requirement

In this project, you are required to do spatial hot spot analysis. In particular, you need to complete two different hot spot analysis tasks.

1. Hot zone analysis

This task will need to perform a range join operation on a rectangle datasets and a point dataset. For each rectangle, the number of points located within the rectangle will be obtained. The hotter rectangle means that it includes more points. So this task is to calculate the hotness of all the rectangles.

Download the required templates below.



2. Hot cell analysis

Description

This task will focus on applying spatial statistics to spatio-temporal big data in order to identify statistically significant spatial hot spots using Apache Spark. The topic of this task is from ACM SIGSPATIAL GISCUUP 2016.

Extract it → then open

C:\yourpath\CSE511-Project-Hotspot-Analysis-20200804T194129Z-001\CSE511-Project-Hotspot-Analysis\src\main\scala\cse512

A- **Hot zone analysis:**

Input is set of rectangles and set of points => aim finding the hotness zone/cell base on number of points in each rectangle.

Function **ST-Contains**

Input 2 strings

- 1) Corner (opposite) points of rectangle
- 2) point

Whether point is in rectangle or not?


essing at Scale - Fall B 2021 Private LIVE — October 11, 2021 - J Edit Item Edit Course Help

> Week 8 > Project 2: Hotspot Analysis - Autograder 2.0 Previous Next >

1. Hot zone analysis

This task will need to perform a range join operation on a rectangle datasets and a point dataset. For each rectangle, the number of points located within the rectangle will be obtained. The hotter rectangle means that it includes more points. So this task is to calculate the hotness of all the rectangles.

Download the required templates below.







CSE511-Project-Hotspot-Analysis-20200504T19:12:27-001
[Download file](#)

2. Hot cell analysis

Description

Handwritten notes: ST-Contains 2 strings
 (3.1, 8.6, 11.7)
 (2.3, 8.4, 3.7, 10.5)
 (2.3, 8.4, 3.7, 10.5)

CSE511-Project-Hotspot-Analysis > src > main > scala > cse512

<input type="checkbox"/> Name	Date modified
 Entrance.scala	4/18/2018 12:49 PM
 HotcellAnalysis.scala	4/18/2018 12:49 PM
 HotcellUtils.scala	4/18/2018 12:49 PM
 HotzoneAnalysis.scala	4/18/2018 12:49 PM
<input checked="" type="checkbox"/>  HotzoneUtils.scala	4/18/2018 12:49 PM

Need to modify this Scala file to write the function **ST-Contains**.

```

package cse512

object HotzoneUtils {

  def ST_Contains(queryRectangle: String, pointString: String ): Boolean = {
    // YOU NEED TO CHANGE THIS PART
    return true // YOU NEED TO CHANGE THIS PART
  }

  // YOU NEED TO CHANGE THIS PART
}

```

Then open:

CSE511-Project-Hotspot-Analysis > src > main > scala > cse512					Search cse512	
<input type="checkbox"/>	Name	Date modified	Type	Size		
<input type="checkbox"/>	Entrance.scala	4/18/2018 12:49 PM	SCALA File	3 KB		
<input type="checkbox"/>	HotcellAnalysis.scala	4/18/2018 12:49 PM	SCALA File	2 KB		
<input type="checkbox"/>	HotcellUtils.scala	4/18/2018 12:49 PM	SCALA File	2 KB		
<input checked="" type="checkbox"/>	HotzoneAnalysis.scala	4/18/2018 12:49 PM	SCALA File	2 KB		
<input type="checkbox"/>	HotzoneUtils.scala	11/22/2021 11:05 AM	SCALA File	1 KB		

```

// Parse point data formats
spark.udf.register("trim", (string : String)=>(string.replace("(",
 "").replace(")", "")))
pointDf = spark.sql("select trim(_c5) as _c5 from point")
pointDf.createOrReplaceTempView("point")

// Load rectangle data
val rectangleDf =
spark.read.format("com.databricks.spark.csv").option("delimiter", "\t").option("header", "false").load(rectanglePath);
rectangleDf.createOrReplaceTempView("rectangle")

// Join two datasets
spark.udf.register("ST_Contains", (queryRectangle:String,
 pointString:String)=>(HotzoneUtils.ST_Contains(queryRectangle, pointString)))
val joinDf = spark.sql("select rectangle._c0 as rectangle, point._c5 as point
 from rectangle,point where ST_Contains(rectangle._c0,point._c5)")
joinDf.createOrReplaceTempView("joinResult")

// YOU NEED TO CHANGE THIS PART
return joinDf // YOU NEED TO CHANGE THIS PART
}

```

Need to add `.coalesce(1)` to the last query And This function merges all the partitions into a single partition and returns the output.

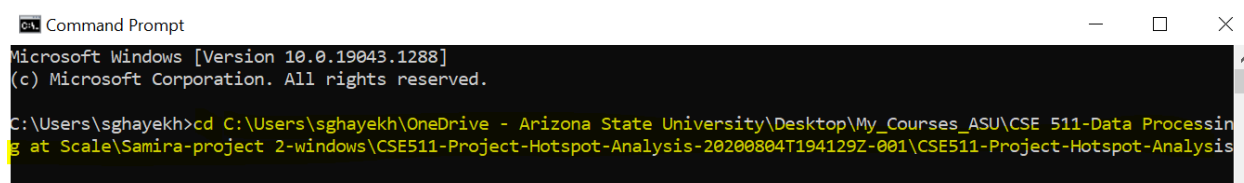
Create a jar file

Now once you wrote the functions in Scala, you need to *create the jar file* and then test it.

Go to the main root of template:

Open the command line: Go to the direction:

Cd C:\YOURPATH\CSE511-Project-Hotspot-Analysis-20200804T194129Z-001\CSE511-Project-Hotspot-Analysis



```

Microsoft Windows [Version 10.0.19043.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Users\sghayekh>cd C:\Users\sghayekh\OneDrive - Arizona State University\Desktop\My_Courses_ASU\CSE 511-Data Processing at Scale\Samira-project 2-windows\CSE511-Project-Hotspot-Analysis-20200804T194129Z-001\CSE511-Project-Hotspot-Analysis

```

Write : **Sbt assembly**

```
C:\Users\sghayekh\OneDrive - Arizona State University\Desktop\My_Courses_ASU\CSE 511-Data Processing at Scale\Samira-project 2-windows\CSE511-Project-Hotspot-Analysis-20200804T194129Z-001\CSE511-Project-Hotspot-Analysis>sbt assembly
```

Takes some time to create a jar file.

```
[info] loading project definition from C:\Users\sghayekh\OneDrive - Arizona State University\Desktop\My_Courses_ASU\CSE 511-Data Processing at Scale\Samira-project 2-windows\CSE511-Project-Hotspot-Analysis-20200804T194129Z-001\CSE511-Project-Hotspot-Analysis
[info] loading settings for project root from build.sbt ...
[info] set current project to CSE512-Hotspot-Analysis-Template (in build file:/C:/Users/sghayekh/Desktop/My_Courses_ASU/CSE%20511-Data%20Processing%20at%20Scale/Samira-project%202-windows/CSE511-Project-Hotspot-Analysis/)
[info] Compiling 5 Scala sources to C:\Users\sghayekh\OneDrive - Arizona State University\Desktop\My_Courses_ASU\CSE 511-Data Processing at Scale\Samira-project 2-windows\CSE511-Project-Hotspot-Analysis-20200804T194129Z-001\CSE511-Project-Hotspot-Analysis\target\scala-2.11\classes ...
[warn] there was one deprecation warning; re-run with -deprecation for details
[warn] one warning found
[info] Including: scala-library-2.11.11.jar
[info] ScalaTest
[info] Run completed in 47 milliseconds.
[info] Total number of tests run: 0
[info] Suites: completed 0, aborted 0
[info] Tests: succeeded 0, failed 0, canceled 0, ignored 0, pending 0
[info] No tests were executed.
[info] Checking every *.class/*.jar file's SHA-1.
[info] Merging files...
[warn] Merging 'META-INF\MANIFEST.MF' with strategy 'discard'
[warn] Strategy 'discard' was applied to a file
[info] SHA-1: f7d1f34c9ea16dfa54fe10fbf07f39f15d15a639
[success] Total time: 13 s, completed Nov 22, 2021 11:45:21 AM
```

See the jar file in

ject-Hotspot-Analysis-20200804T194129Z-001 > CSE511-Project-Hotspot-Analysis				Search CSE511	
<input type="checkbox"/> Name	Date modified	Type	Size		
project	11/22/2021 11:40 AM	File folder			
src	11/22/2021 10:34 AM	File folder			
<input checked="" type="checkbox"/> target	11/22/2021 11:40 AM	File folder			
testcase	11/22/2021 10:34 AM	File folder			
.gitignore	4/18/2018 12:49 PM	GITIGNORE File	1 KB		
build.sbt	4/18/2018 12:49 PM	SBT File	1 KB		
README.md	8/4/2020 11:30 AM	MD File	7 KB		

SE511-Project-Hotspot-Analysis > target > scala-2.11

Name	Date modified	Type	Size
classes	11/22/2021 11:45 AM	File folder	
update	11/22/2021 11:40 AM	File folder	
<input checked="" type="checkbox"/> CSE512-Hotspot-Analysis-Template-assembl...	11/22/2021 11:45 AM	Executable Jar File	5,640 KB

Now change this part in resource:

« CSE511-Project-Hotspot-Analysis > src > resources

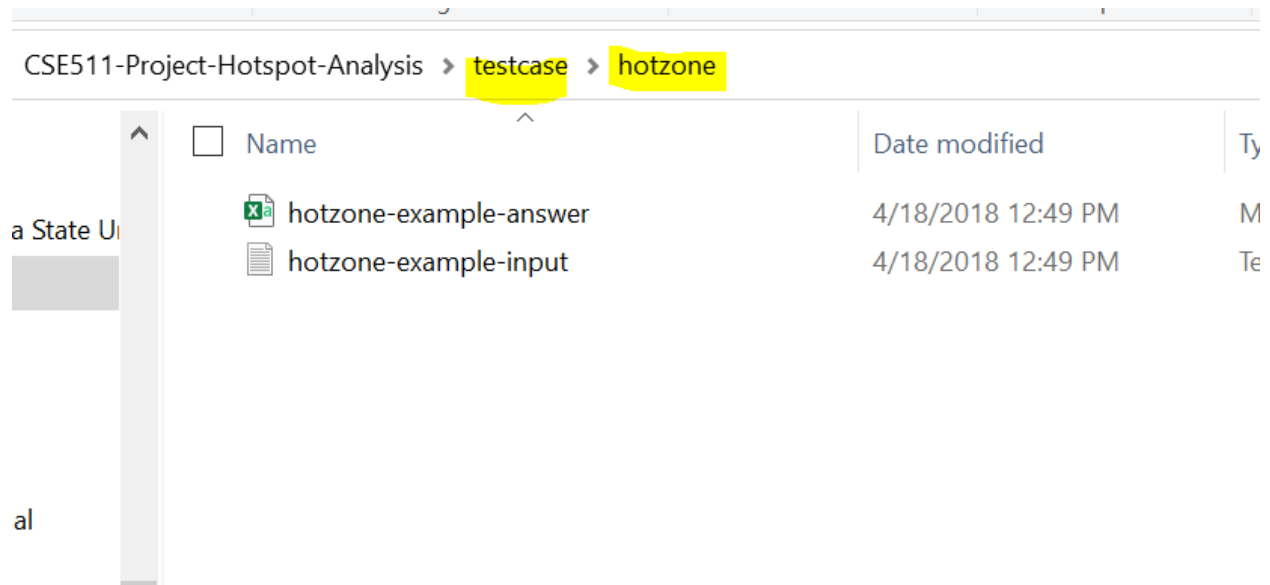
Name	Date modified	Type
.gitignore	4/18/2018 12:49 PM	GITIGNORE File
point-hotzone	4/18/2018 12:49 PM	Microsoft Excel Con
yellow_trip_sample_100000	4/18/2018 12:49 PM	Microsoft Excel Con
zone-hotzone	4/18/2018 12:49 PM	Microsoft Excel Con

- to -

Test your code

After creating a jar file need to test it.

Now test your Jar file with input test cases:



Make sure you have a right slash (/):

Forward slash set used for (Linux). **Windows** use **back slash**.

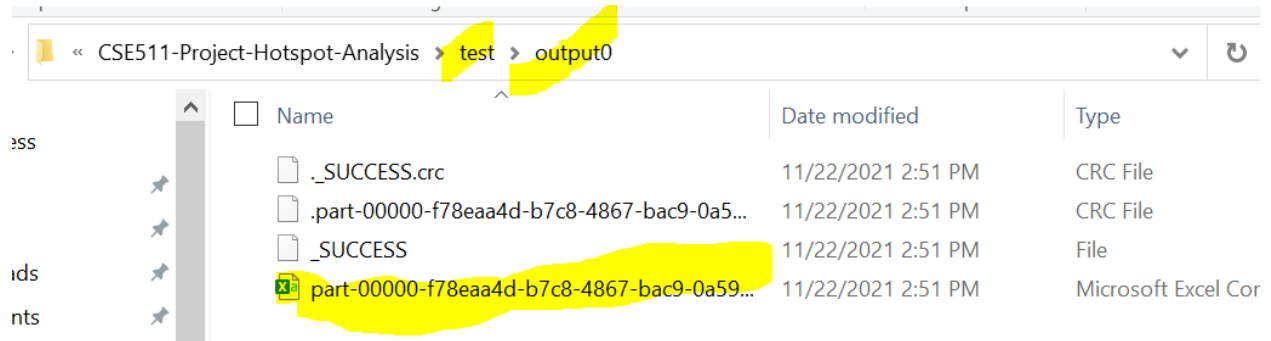
//windows

test\output hotzoneanalysis src\resources\point-hotzone.csv src\resources\zone-hotzone.csv

Need to write this part as well:

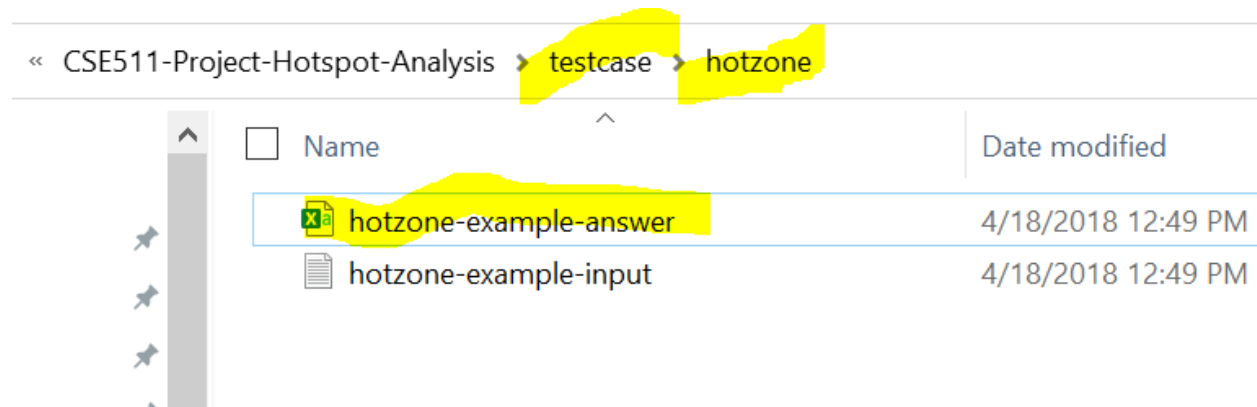
```
C:\Users\sghayekh\OneDrive - Arizona State University\Desktop\My_Courses_ASU\CSE 511-Data Processing at Scale\Samira-project 2-windows\CSE511-Project-Hotspot-Analysis-20200804T194129Z-001\CSE511-Project-Hotspot-Analysis>spark-submit target\scala-2.11\CSE512-Hotspot-Analysis-Template-assembly-0.1.0.jar test\output hotzoneanalysis src\resources\point-hotzone.csv src\resources\zone-hotzone.csv
```


If everything were correct you will see the output in the following path:



Name	Date modified	Type
._SUCCESS.crc	11/22/2021 2:51 PM	CRC File
.part-00000-f78eaa4d-b7c8-4867-bac9-0a5...	11/22/2021 2:51 PM	CRC File
._SUCCESS	11/22/2021 2:51 PM	File
part-00000-f78eaa4d-b7c8-4867-bac9-0a59...	11/22/2021 2:51 PM	Microsoft Excel Cor

Its content must be the same as output (hot zone-example-answer) as we have in template:



Name	Date modified
hotzone-example-answer	4/18/2018 12:49 PM
hotzone-example-input	4/18/2018 12:49 PM

Done!

Hot Cell analysis:

1. After applying changes in "HotcellAnalysis.scala and HotcellUtils.scala"
2. Then make a Jar file.

```
C:\Users\sghayekh\OneDrive - Arizona State University\Desktop\My_Courses_ASU\CSE 511-Data Processing at Scale\Samira-
ject 2-windows\CSE511-Project-Hotspot-Analysis-20200804T194129Z-001\CSE511-Project-Hotspot-Analysis>Sbt assembly
[info] welcome to sbt 1.3.13 (Amazon.com Inc. Java 1.8.0_312)
[info] loading settings for project cse511-project-hotspot-analysis-build from plugins.sbt ...
```

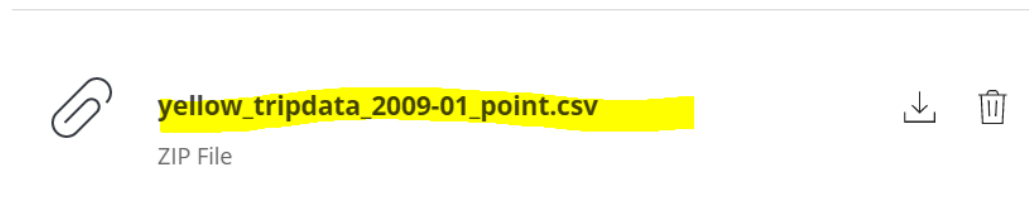
Jar file created.

Before testing, We also need to download the point data from Coursera and paste it in the following path before test the jar file.

Input data format

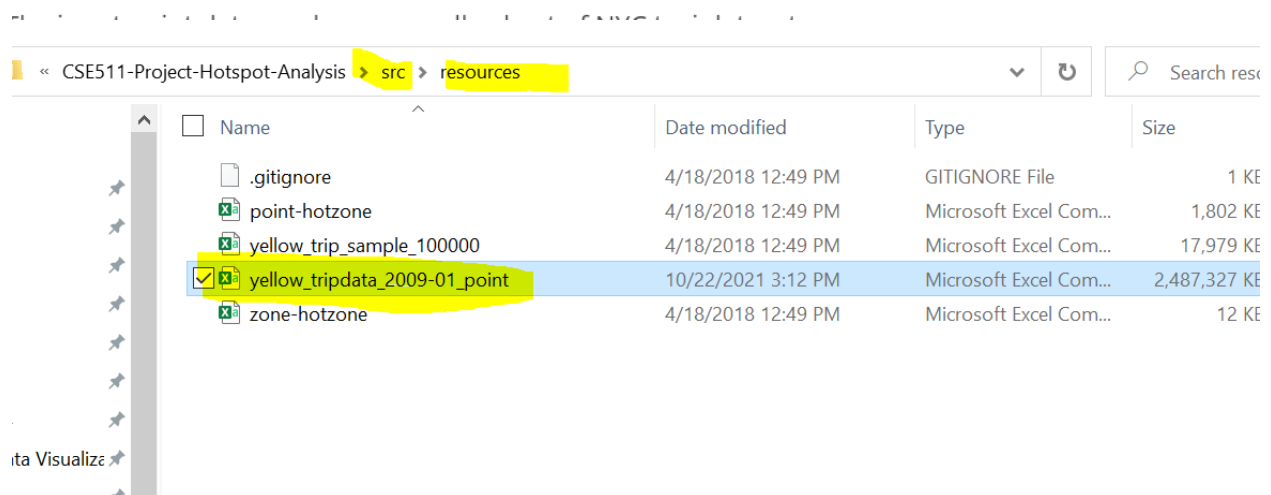
The main function/entrance is "cse512.Entrance" scala file.

1. Point data: the input point dataset is the pickup point of New York Taxi trip datasets. But the coding template already parsed it for you. Find the data in the .zip file below.

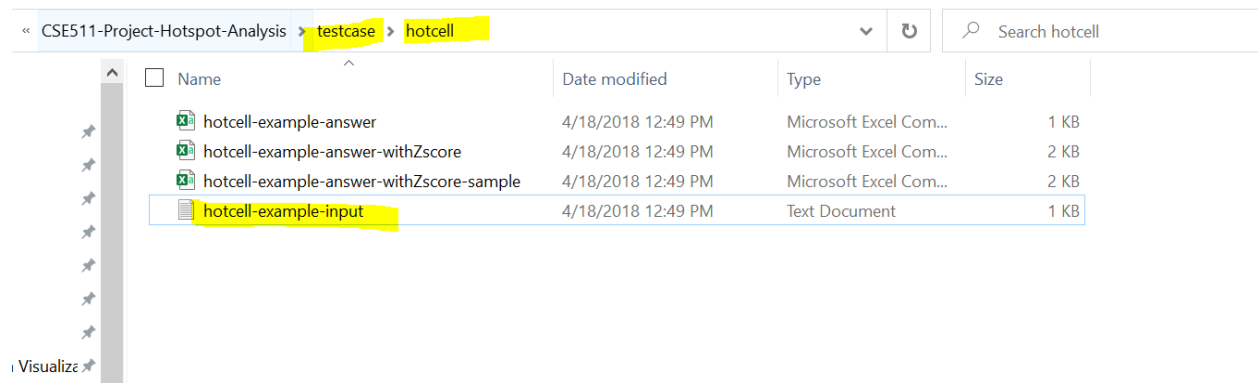


2. Zone data (only for hot zone analysis): at "src/resources/zone-hotzone" of the template

Hot zone analysis



Then test the test cases from the following path:



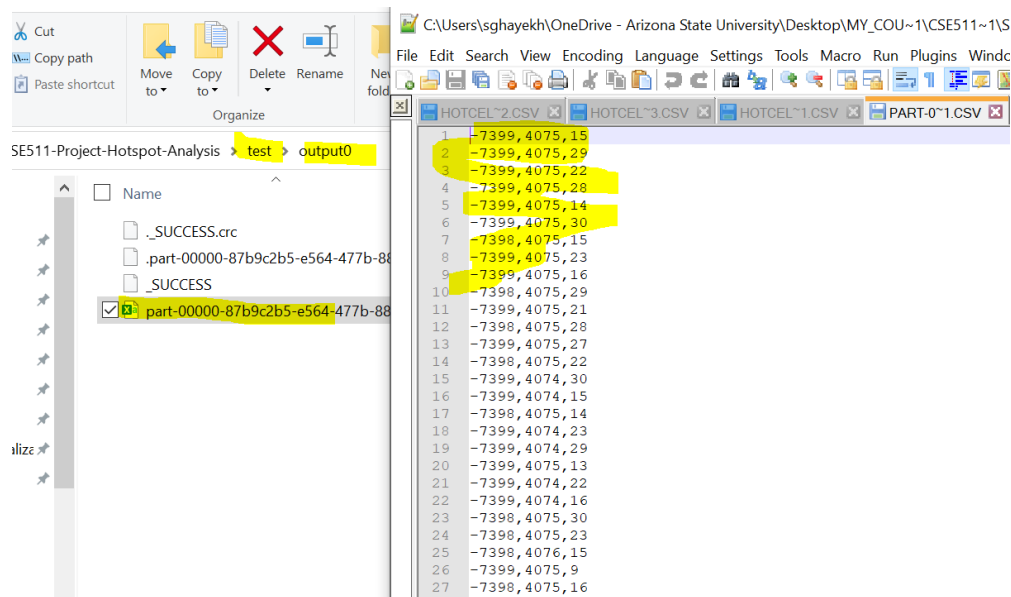
//windows

test\output hotcellanalysis src\resources\yellow_tripdata_2009-01_point.csv

Need to write the following commands:

```
C:\Users\sghayekh\OneDrive - Arizona State University\Desktop\My_Courses_ASU\CSE 511-Data Processing at Scale\Samira-project 2-windows\CSE511-Project-Hotspot-Analysis-20200804T194129Z-001\CSE511-Project-Hotspot-Analysis>spark-submit target\scala-2.11\CSE512-Hotspot-Analysis-Template-assembly-0.1.0.jar test\output hotcellanalysis src\resources\yellow_tripdata_2009-01_point.csv
```

Here is the output in test folder



Done!

And my jar file passed the auto grader successfully.