DATADRIVEN FRAMEWORK

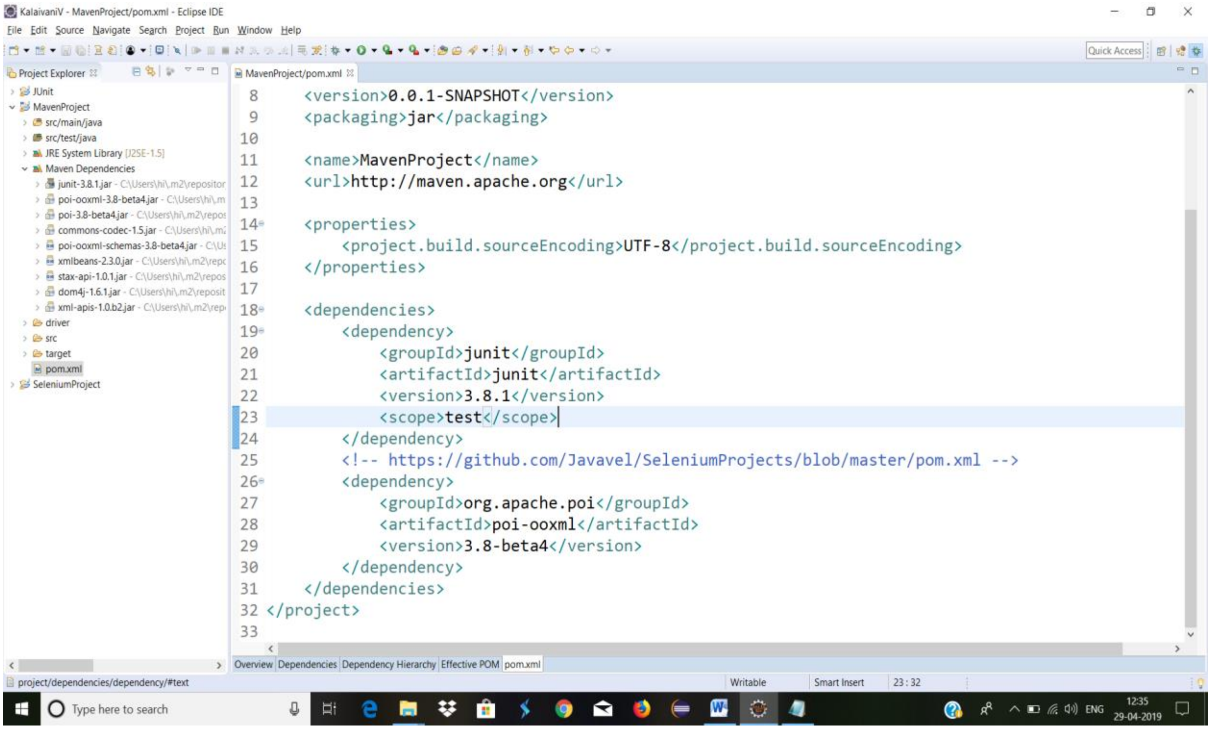
- Data Driven Framework is used to read the value from excel sheet.

TYPES OF EXCEL FORMAT

.XLS	.XLSX
Row: 2^16	Row: 2^20
Column: 2^8	Column: 2^14
Jar Name: JXL POI	Jar Name: POI

STEPS:

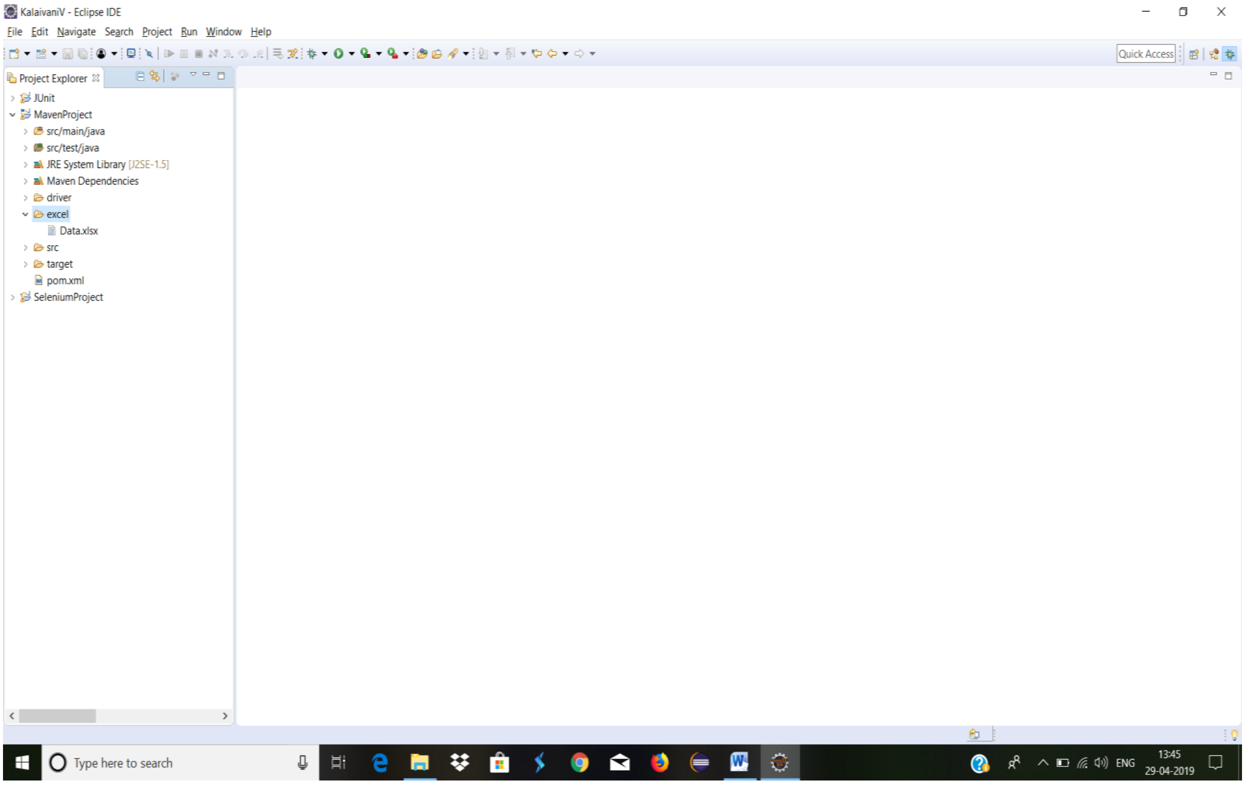
- Create the Maven Project
- Copy the Maven dependency for data driven framework and paste in pom.xml file.
  “Poi-ooxml.3.8-beta4”.



KalaivaniV - MavenProject/pom.xml - Eclipse IDE
File Edit Source Navigate Search Project Run Window Help
Quick Access
Project Explorer
JUnit
MavenProject
src/main/java
src/test/java
JRE System Library [J2SE-1.5]
Maven Dependencies
junit-3.8.1.jar - C:\Users\hi\.m2\repositor
poi-ooxml-3.8-beta4.jar - C:\Users\hi\.m
poi-3.8-beta4.jar - C:\Users\hi\.m2\repos
commons-codec-1.5.jar - C:\Users\hi\.m
poi-ooxml-schemas-3.8-beta4.jar - C:\U
xmlbeans-2.3.0.jar - C:\Users\hi\.m2\rep
stax-api-1.0.1.jar - C:\Users\hi\.m2\repos
dom4j-1.6.1.jar - C:\Users\hi\.m2\reposit
xml-apis-1.0.b2.jar - C:\Users\hi\.m2\rep
driver
src
target
pom.xml
SeleniumProject
MavenProject/pom.xml
Overview Dependencies Dependency Hierarchy Effective POM pom.xml
project/dependencies/dependency/#text
Writable
Smart Insert
23 : 32
Type here to search
ENG
12:35
29-04-2019

```
    <version>0.0.1-SNAPSHOT</version>
    <packaging>jar</packaging>

    <name>MavenProject</name>
    <url>http://maven.apache.org</url>

    <properties>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    </properties>

    <dependencies>
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>3.8.1</version>
            <scope>test</scope>
        </dependency>
        <!-- https://github.com/Javavel/SeleniumProjects/blob/master/pom.xml -->
        <dependency>
            <groupId>org.apache.poi</groupId>
            <artifactId>poi-ooxml</artifactId>
            <version>3.8-beta4</version>
        </dependency>
    </dependencies>
</project>

```

- ➢ Create one folder for excel and paste the excel sheet inside the excel folder.



- ➢ Create the normal class and write down the program.

DEPENDENCY:

```
<dependency>
        <groupId>org.apache.poi</groupId>
        <artifactId>poi-ooxml</artifactId>
        <version>3.8-beta4</version>
</dependency>
```

TYPES OF WORKBOOK:

- ➢ HSSFWorkbook - .XLS
- ➢ XSSFWorkbook - .XLSX

Order of the program:

Row and Cell will start from 0 to n-1

- Set the file location
  File loc=new File("Excel file location");

- We are reading the inputs from the excel so class name is "FileInputStream"
  FileInputStream s=new FileInputStream(File location variable name);

- Set the Workbook name
  Workbook w=new XSSFWorkbook(FileInputStream variable name);

- Select the sheet name
  Sheet s=w.getSheet("Sheetname");

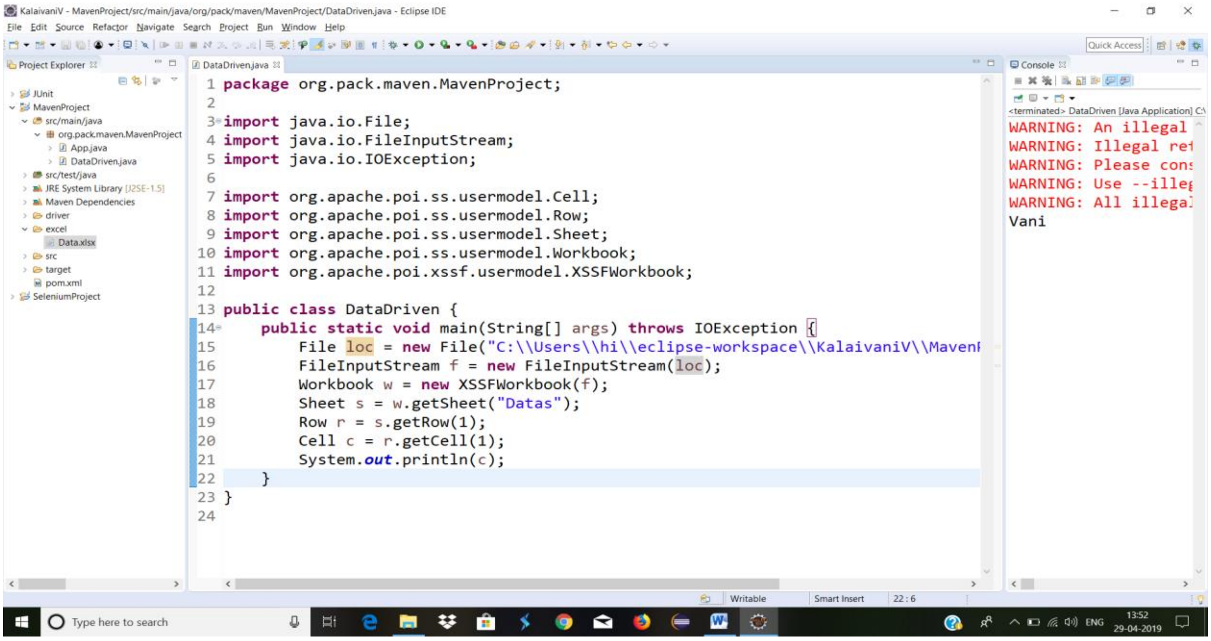
- Select Row
  Row r=s.getRow(row number);

- Select Cell
  Cell c=r.getCell(cell number)

Interface Names:

- Workbook
- Sheet
- Row
- Cell

How to read particular cell value from the excel:



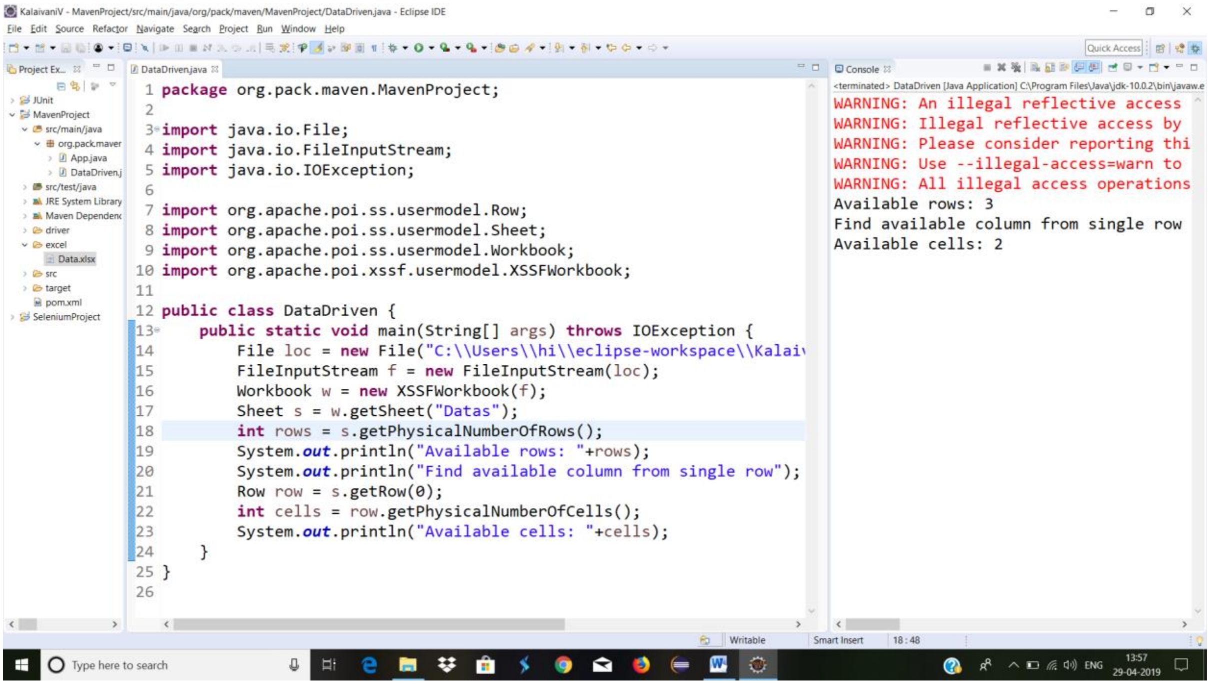
KalaivaniV - MavenProject/src/main/java/org/pack/maven/MavenProject/DataDriven.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
DataDriven.java
Console
Writable
Smart Insert
22 : 6

```java
package org.pack.maven.MavenProject;

import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;

import org.apache.poi.ss.usermodel.Cell;
import org.apache.poi.ss.usermodel.Row;
import org.apache.poi.ss.usermodel.Sheet;
import org.apache.poi.ss.usermodel.Workbook;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;

public class DataDriven {
    public static void main(String[] args) throws IOException {
        File loc = new File("C:\\Users\\hi\\eclipse-workspace\\KalaivaniV\\MavenP
        FileInputStream f = new FileInputStream(loc);
        Workbook w = new XSSFWorkbook(f);
        Sheet s = w.getSheet("Datas");
        Row r = s.getRow(1);
        Cell c = r.getCell(1);
        System.out.println(c);
    }
}
```

```
<terminated> DataDriven [Java Application] C:\
WARNING: An illegal
WARNING: Illegal ref
WARNING: Please cons
WARNING: Use --illeg
WARNING: All illegal
Vani
```

How to find the available rows and how to find available columns from single row:

- ➢ getPhysicalNumberOfRows: used to get available rows
- ➢ getPhysicalNumberOfCells: used to get available cells

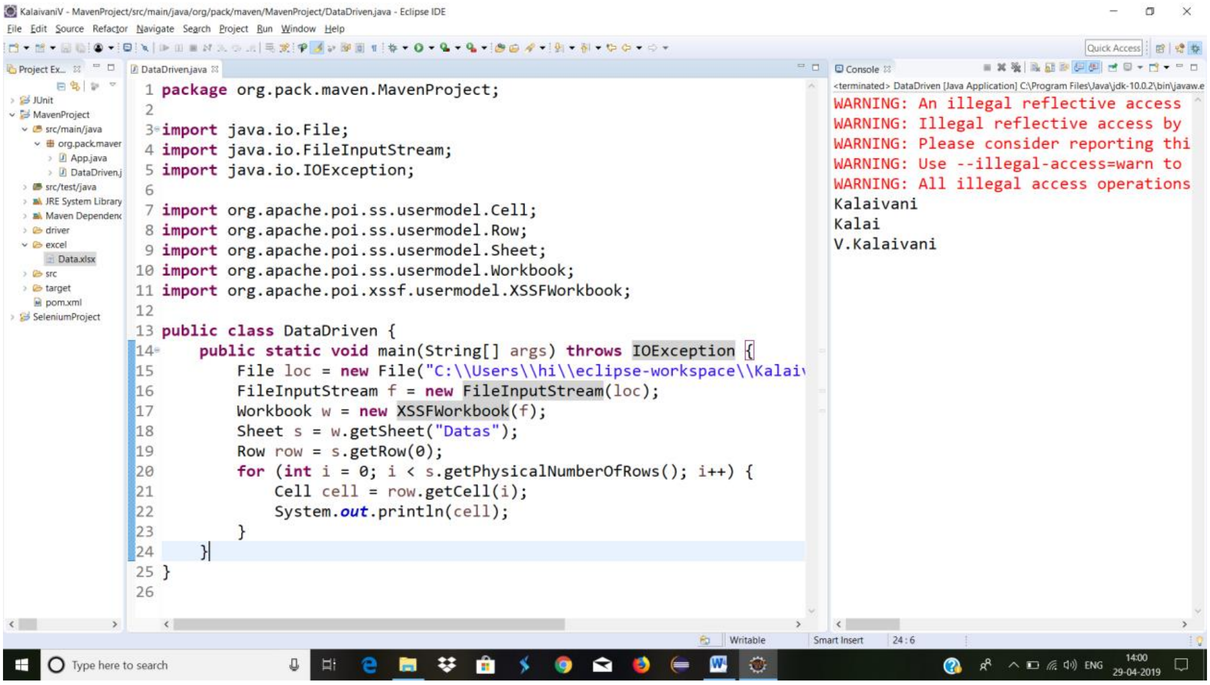


KalaivaniV - MavenProject/src/main/java/org/pack/maven/MavenProject/DataDriven.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Ex...
DataDriven.java
Console
Writable
Smart Insert
18 : 48

```java
package org.pack.maven.MavenProject;

import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;

import org.apache.poi.ss.usermodel.Row;
import org.apache.poi.ss.usermodel.Sheet;
import org.apache.poi.ss.usermodel.Workbook;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;

public class DataDriven {
    public static void main(String[] args) throws IOException {
        File loc = new File("C:\\Users\\hi\\eclipse-workspace\\Kalaiv
        FileInputStream f = new FileInputStream(loc);
        Workbook w = new XSSFWorkbook(f);
        Sheet s = w.getSheet("Datas");
        int rows = s.getPhysicalNumberOfRows();
        System.out.println("Available rows: "+rows);
        System.out.println("Find available column from single row");
        Row row = s.getRow(0);
        int cells = row.getPhysicalNumberOfCells();
        System.out.println("Available cells: "+cells);
    }
}
```

```
<terminated> DataDriven [Java Application] C:\Program Files\Java\jdk-10.0.2\bin\javaw.e
WARNING: An illegal reflective access
WARNING: Illegal reflective access by
WARNING: Please consider reporting thi
WARNING: Use --illegal-access=warn to
WARNING: All illegal access operations
Available rows: 3
Find available column from single row
Available cells: 2
```

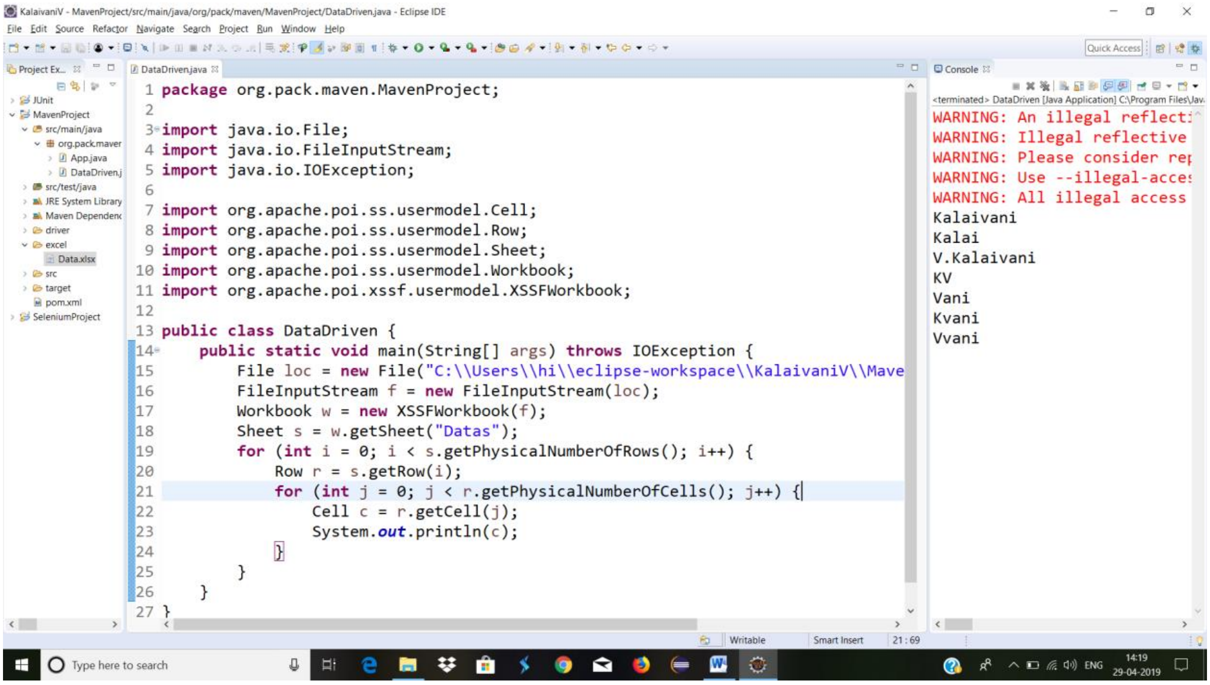
How to print all the value from particular row:



```
package org.pack.maven.MavenProject;

import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;

import org.apache.poi.ss.usermodel.Cell;
import org.apache.poi.ss.usermodel.Row;
import org.apache.poi.ss.usermodel.Sheet;
import org.apache.poi.ss.usermodel.Workbook;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;

public class DataDriven {
    public static void main(String[] args) throws IOException {
        File loc = new File("C:\\Users\\hi\\eclipse-workspace\\Kalai
        FileInputStream f = new FileInputStream(loc);
        Workbook w = new XSSFWorkbook(f);
        Sheet s = w.getSheet("Datas");
        Row row = s.getRow(0);
        for (int i = 0; i < s.getPhysicalNumberOfRows(); i++) {
            Cell cell = row.getCell(i);
            System.out.println(cell);
        }
    }
}
```

```
<terminated> DataDriven [Java Application] C:\Program Files\Java\jdk-10.0.2\bin\javaw.e
WARNING: An illegal reflective access
WARNING: Illegal reflective access by
WARNING: Please consider reporting thi
WARNING: Use --illegal-access=warn to
WARNING: All illegal access operations
Kalaivani
Kalai
V.Kalaivani
```

How to print all the value from row and cells



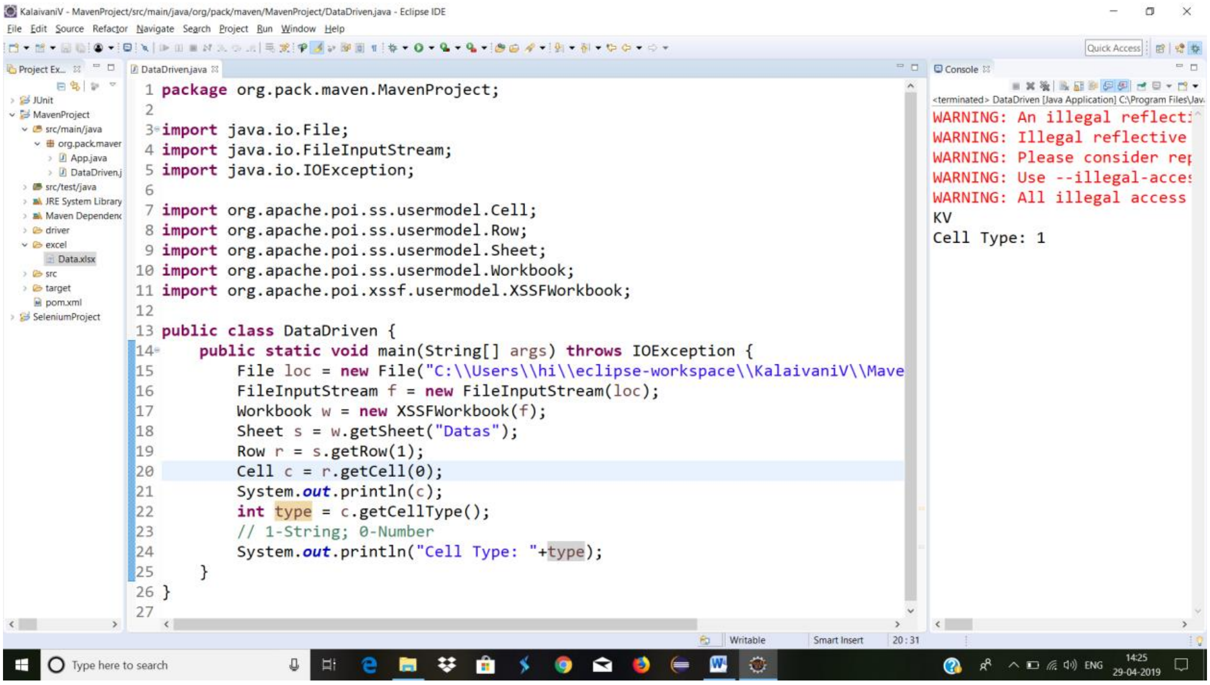
```
package org.pack.maven.MavenProject;

import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;

import org.apache.poi.ss.usermodel.Cell;
import org.apache.poi.ss.usermodel.Row;
import org.apache.poi.ss.usermodel.Sheet;
import org.apache.poi.ss.usermodel.Workbook;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;

public class DataDriven {
    public static void main(String[] args) throws IOException {
        File loc = new File("C:\\Users\\hi\\eclipse-workspace\\KalaivaniV\\Mave
        FileInputStream f = new FileInputStream(loc);
        Workbook w = new XSSFWorkbook(f);
        Sheet s = w.getSheet("Datas");
        for (int i = 0; i < s.getPhysicalNumberOfRows(); i++) {
            Row r = s.getRow(i);
            for (int j = 0; j < r.getPhysicalNumberOfCells(); j++) {
                Cell c = r.getCell(j);
                System.out.println(c);
            }
        }
    }
}
```

```
<terminated> DataDriven [Java Application] C:\Program Files\Jav
WARNING: An illegal reflect:
WARNING: Illegal reflective
WARNING: Please consider rep
WARNING: Use --illegal-acce:
WARNING: All illegal access
Kalaivani
Kalai
V.Kalaivani
KV
Vani
Kvani
Vvani
```

Enhanced for loop can’t be used here, because there is no data type.

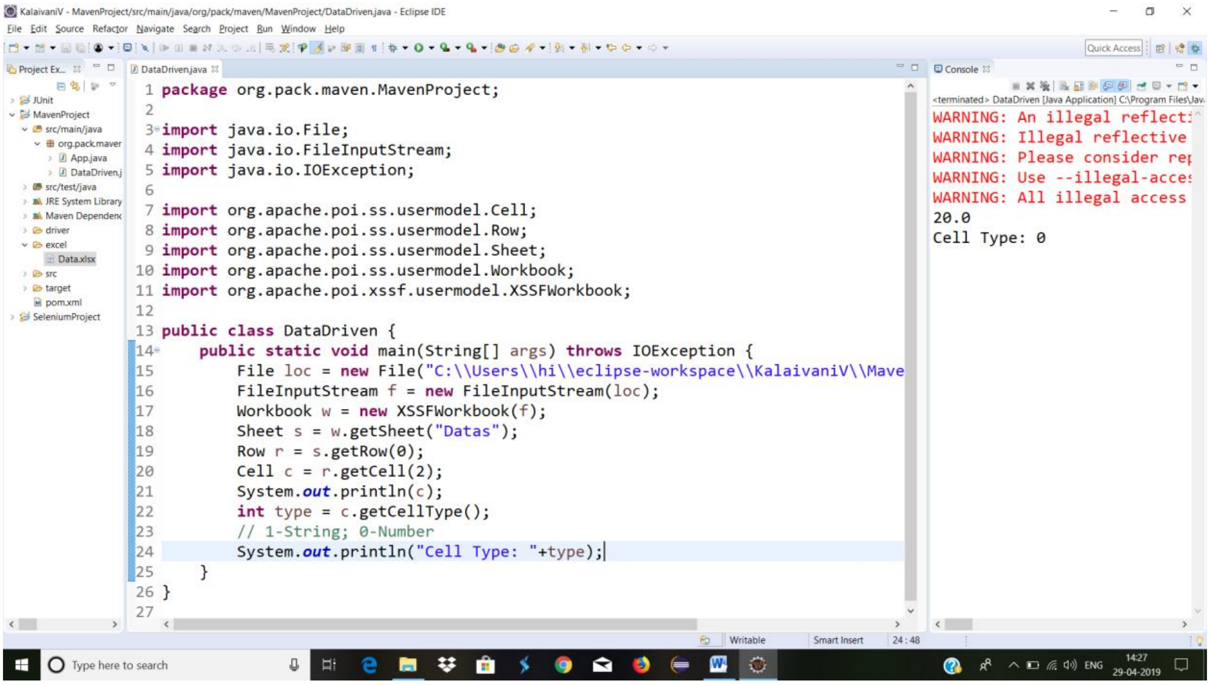
How to find the cell type:

If cell have String mean the output is 1
If cell have Number, Date mean the output is 0



```java
package org.pack.maven.MavenProject;

import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;

import org.apache.poi.ss.usermodel.Cell;
import org.apache.poi.ss.usermodel.Row;
import org.apache.poi.ss.usermodel.Sheet;
import org.apache.poi.ss.usermodel.Workbook;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;

public class DataDriven {
    public static void main(String[] args) throws IOException {
        File loc = new File("C:\\Users\\hi\\eclipse-workspace\\KalaivaniV\\Mave
        FileInputStream f = new FileInputStream(loc);
        Workbook w = new XSSFWorkbook(f);
        Sheet s = w.getSheet("Datas");
        Row r = s.getRow(1);
        Cell c = r.getCell(0);
        System.out.println(c);
        int type = c.getCellType();
        // 1-String; 0-Number
        System.out.println("Cell Type: "+type);
    }
}
```

```
<terminated> DataDriven [Java Application] C:\Program Files\Jav
WARNING: An illegal reflect
WARNING: Illegal reflective
WARNING: Please consider rep
WARNING: Use --illegal-acces
WARNING: All illegal access
KV
Cell Type: 1
```



```java
package org.pack.maven.MavenProject;

import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;

import org.apache.poi.ss.usermodel.Cell;
import org.apache.poi.ss.usermodel.Row;
import org.apache.poi.ss.usermodel.Sheet;
import org.apache.poi.ss.usermodel.Workbook;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;

public class DataDriven {
    public static void main(String[] args) throws IOException {
        File loc = new File("C:\\Users\\hi\\eclipse-workspace\\KalaivaniV\\Mave
        FileInputStream f = new FileInputStream(loc);
        Workbook w = new XSSFWorkbook(f);
        Sheet s = w.getSheet("Datas");
        Row r = s.getRow(0);
        Cell c = r.getCell(2);
        System.out.println(c);
        int type = c.getCellType();
        // 1-String; 0-Number
        System.out.println("Cell Type: "+type);
    }
}
```

```
<terminated> DataDriven [Java Application] C:\Program Files\Jav
WARNING: An illegal reflect
WARNING: Illegal reflective
WARNING: Please consider rep
WARNING: Use --illegal-acces
WARNING: All illegal access
20.0
Cell Type: 0
```

How to get string, date, and number value from the excel:

- getStringCellValue()
- getDateCellValue()
- getNumericCellValue()

Get String Value:

```
if (type == 1) {
    String string = c.getStringCellValue();
    System.out.println(string);
}
```

Get Date Value:

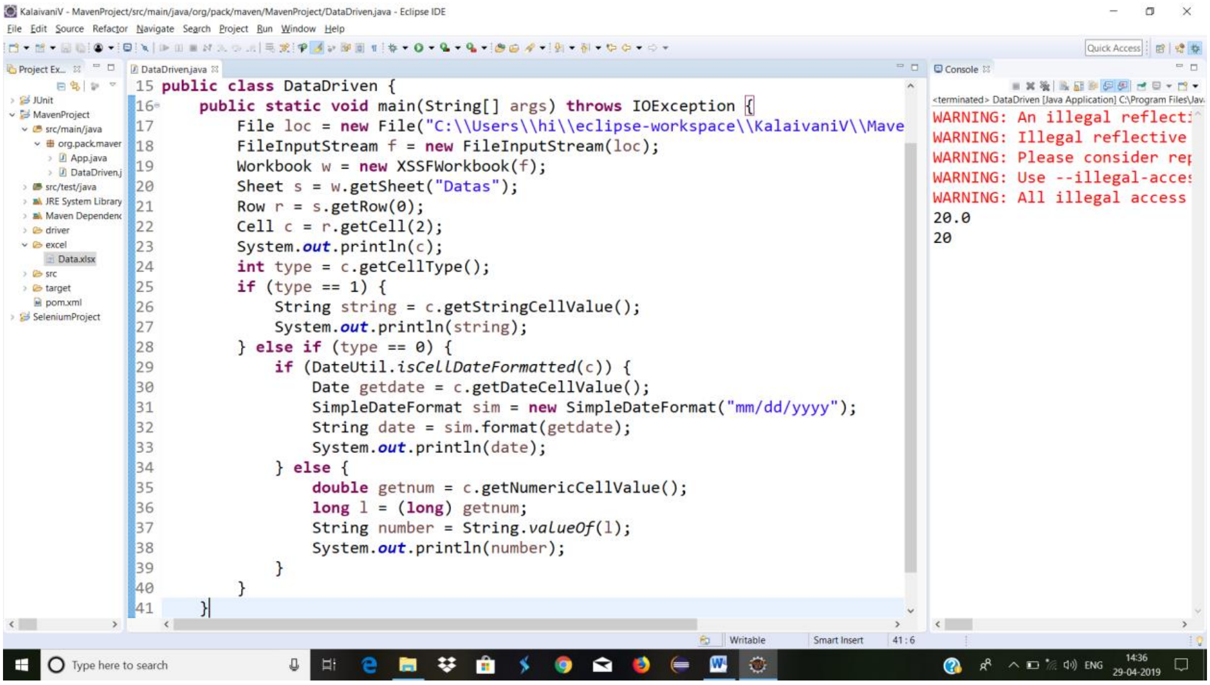
Format is the method is used to convert date to string.

```
else if (type == 0) {
    if (DateUtil.isCellDateFormatted(c)) {
    Date getdate = c.getDateCellValue();
    SimpleDateFormat sim = new
SimpleDateFormat("mm/dd/yyyy");
    String date = sim.format(getdate);
    System.out.println(date);
}
```

Get Numeric Value:

ValueOf is the method is used to covert Number to String.

```
else {
    double getnum =c.getNumericCellValue();
    long l = (long) getnum;
    String number = String.valueOf(l);
    System.out.println(number);
}
```

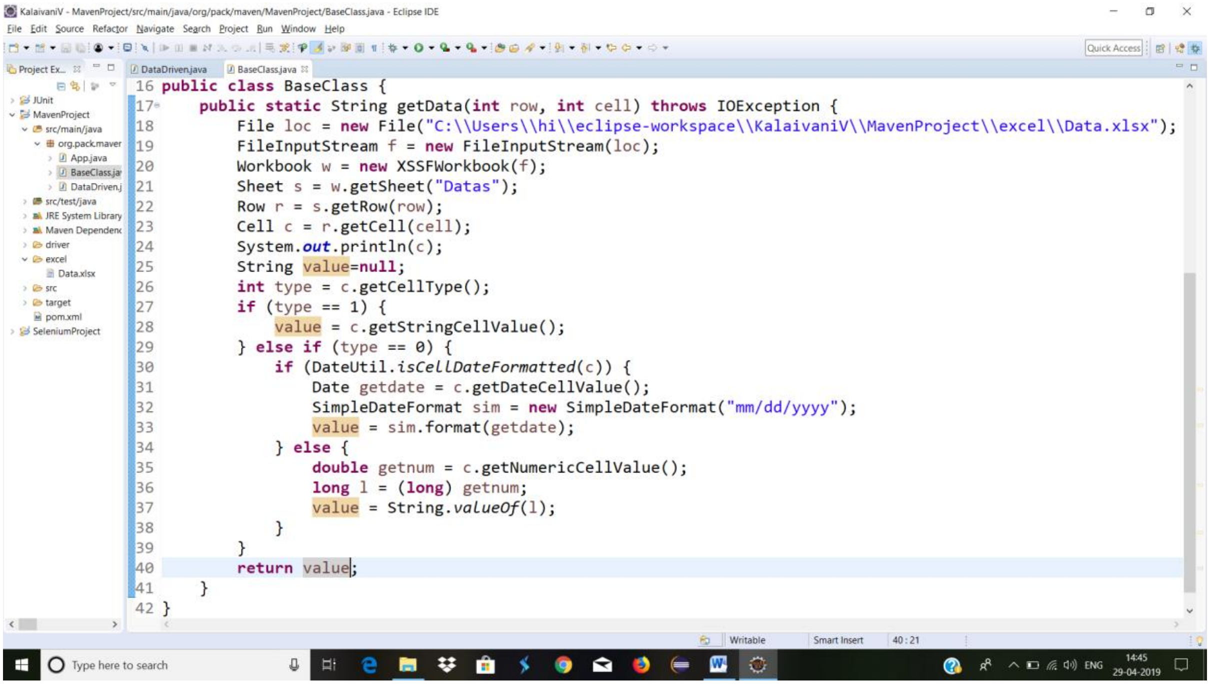

KalaivaniV - MavenProject/src/main/java/org/pack/maven/MavenProject/DataDriven.java - Eclipse IDE
DataDriven.java
Console

```java
public class DataDriven {
    public static void main(String[] args) throws IOException {
        File loc = new File("C:\\Users\\hi\\eclipse-workspace\\KalaivaniV\\Mave
        FileInputStream f = new FileInputStream(loc);
        Workbook w = new XSSFWorkbook(f);
        Sheet s = w.getSheet("Datas");
        Row r = s.getRow(0);
        Cell c = r.getCell(2);
        System.out.println(c);
        int type = c.getCellType();
        if (type == 1) {
            String string = c.getStringCellValue();
            System.out.println(string);
        } else if (type == 0) {
            if (DateUtil.isCellDateFormatted(c)) {
                Date getdate = c.getDateCellValue();
                SimpleDateFormat sim = new SimpleDateFormat("mm/dd/yyyy");
                String date = sim.format(getdate);
                System.out.println(date);
            } else {
                double getnum = c.getNumericCellValue();
                long l = (long) getnum;
                String number = String.valueOf(l);
                System.out.println(number);
            }
        }
    }
```

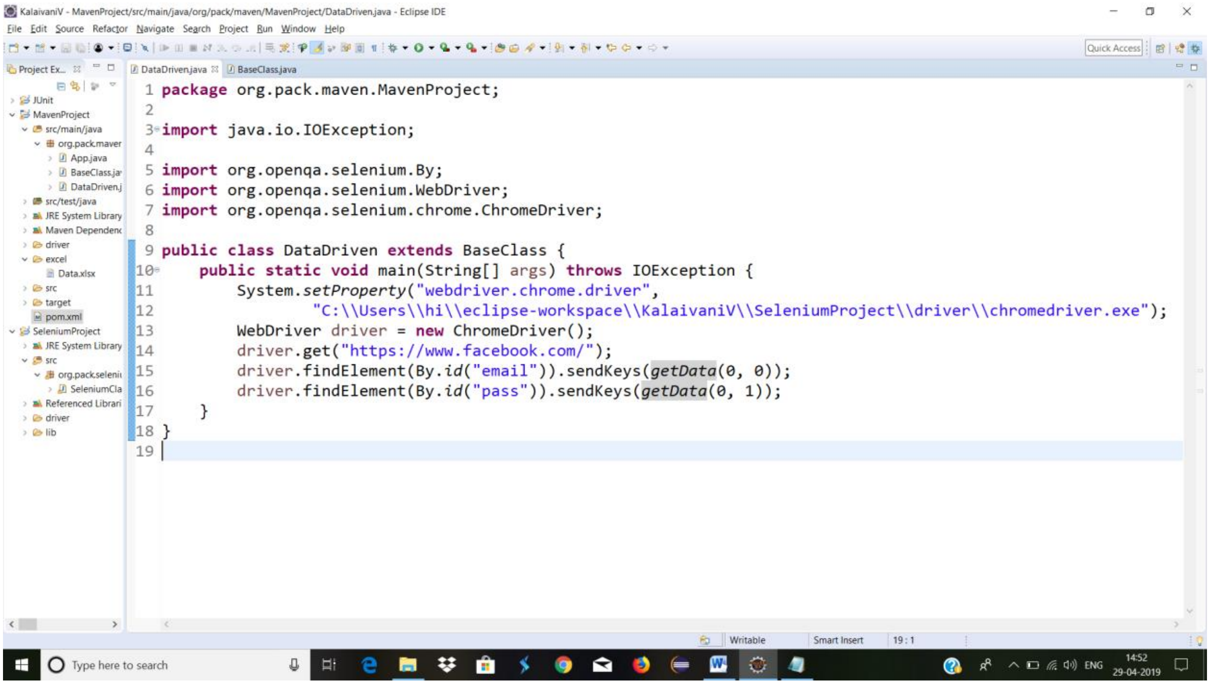
```
<terminated> DataDriven [Java Application] C:\Program Files\Jav
WARNING: An illegal reflecti
WARNING: Illegal reflective
WARNING: Please consider rep
WARNING: Use --illegal-acces
WARNING: All illegal access
20.0
20
```

How to integrate with selenium:

If we want to send the value from excel mean, each and every time need create the program because row and cell will be changed instead of creating the program again we are creating the program as a one method into base class. If we want to get any value just pass only row and cell.


KalaivaniV - MavenProject/src/main/java/org/pack/maven/MavenProject/BaseClass.java - Eclipse IDE
DataDriven.java
BaseClass.java

```java
public class BaseClass {
    public static String getData(int row, int cell) throws IOException {
        File loc = new File("C:\\Users\\hi\\eclipse-workspace\\KalaivaniV\\MavenProject\\excel\\Data.xlsx");
        FileInputStream f = new FileInputStream(loc);
        Workbook w = new XSSFWorkbook(f);
        Sheet s = w.getSheet("Datas");
        Row r = s.getRow(row);
        Cell c = r.getCell(cell);
        System.out.println(c);
        String value=null;
        int type = c.getCellType();
        if (type == 1) {
            value = c.getStringCellValue();
        } else if (type == 0) {
            if (DateUtil.isCellDateFormatted(c)) {
                Date getdate = c.getDateCellValue();
                SimpleDateFormat sim = new SimpleDateFormat("mm/dd/yyyy");
                value = sim.format(getdate);
            } else {
                double getnum = c.getNumericCellValue();
                long l = (long) getnum;
                value = String.valueOf(l);
            }
        }
        return value;
    }
}
```



```java
package org.pack.maven.MavenProject;

import java.io.IOException;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class DataDriven extends BaseClass {
    public static void main(String[] args) throws IOException {
        System.setProperty("webdriver.chrome.driver",
                "C:\\Users\\hi\\eclipse-workspace\\KalaivaniV\\SeleniumProject\\driver\\chromedriver.exe");
        WebDriver driver = new ChromeDriver();
        driver.get("https://www.facebook.com/");
        driver.findElement(By.id("email")).sendKeys(getData(0, 0));
        driver.findElement(By.id("pass")).sendKeys(getData(0, 1));
    }
}

```