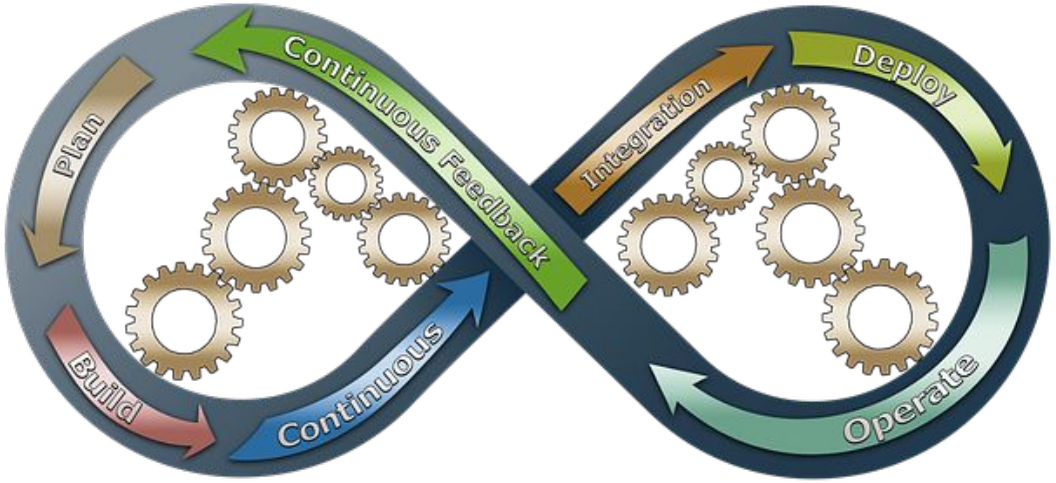


Session 4.9

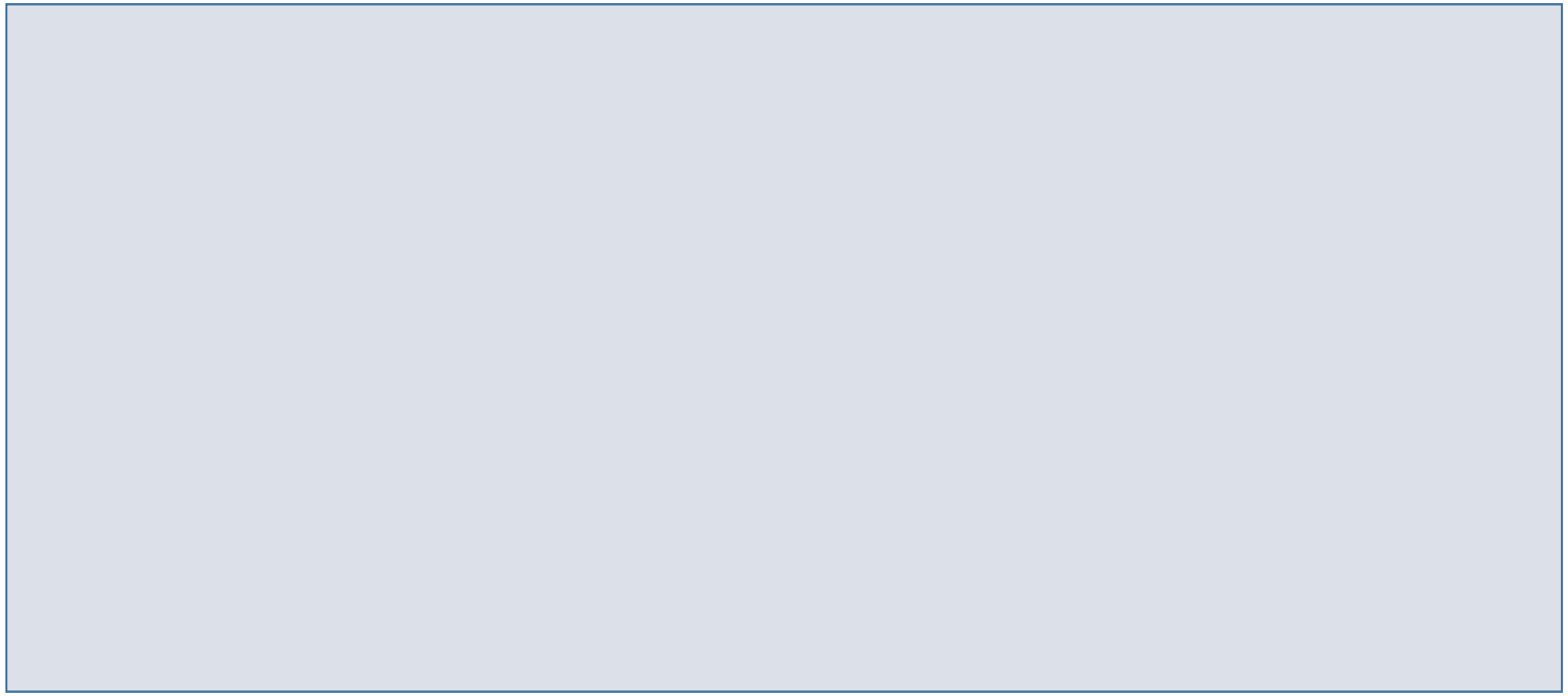
Apache POI

AN INITIATIVE BY

UNICAL ACADEMY

Introduction

UNICAL ACADEMY



Let’s go!!!

UNICAL ACADEMY

Apache POI

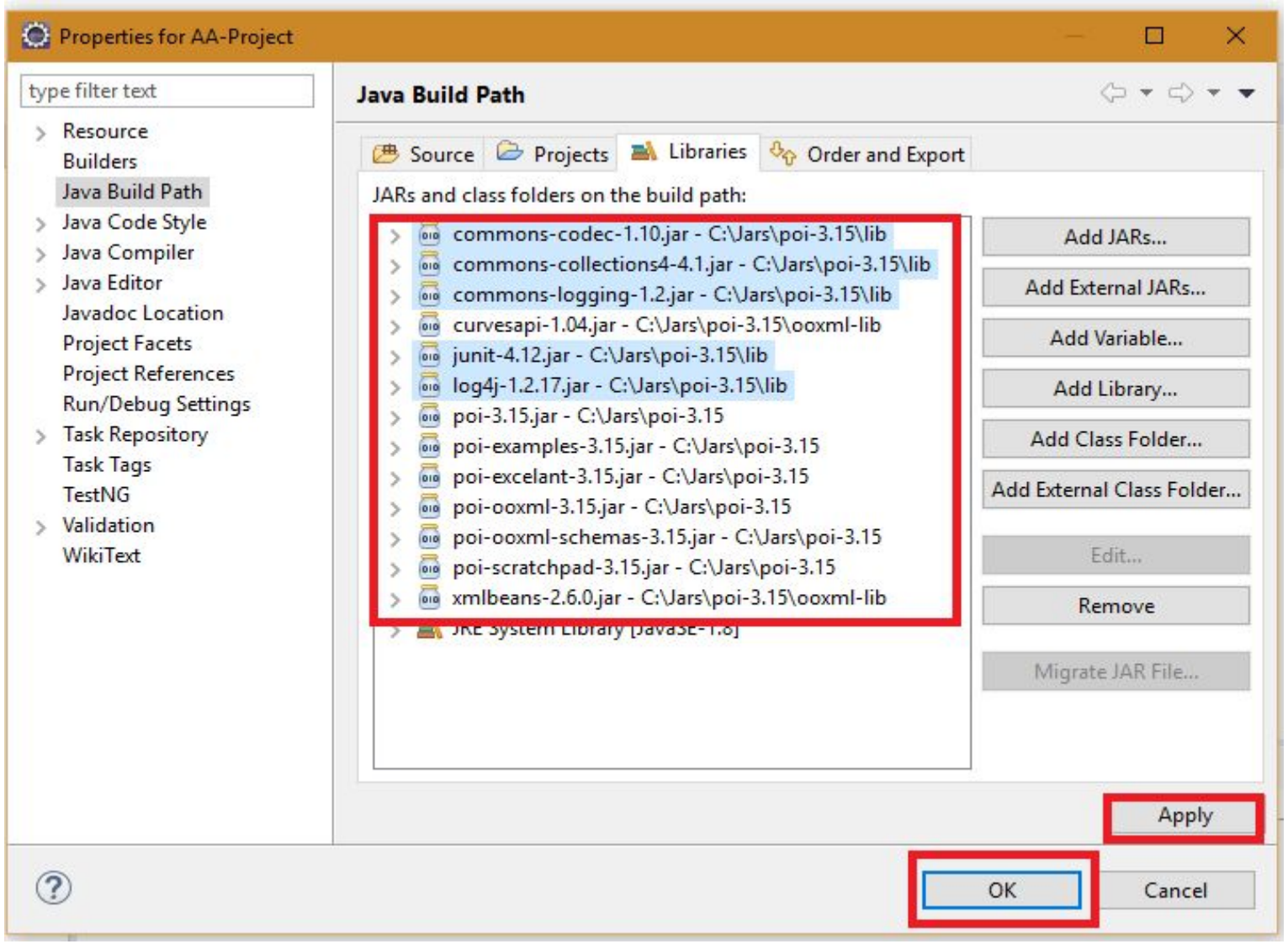
- **Step1: Download Apache POI.** You can download Apache POI from its ***official website*** (https://poi.apache.org/download.html).

- **Step2: Extract the downloaded POI jar files.** Once zip file is downloaded from above link, extract the jar files to some location.

- **Step 3: Create new Java project.**

- **Step4: Add extracted jar files in to java – selenium project.** Follow below steps to know how can we add the jar files in to a project. First, right click on project created.
    1. Then Go to “Build path” -> “Configure Build Path”
    2. Then select all extracted jar files and click on apply and save.
    3. Then Click on “Add External Jar” option.

UNICAL ACADEMY



Workbook, Sheet Row ,Cell Interfaces methods

Method	Description
CellStyle createCellStyle()	It create a new Cell style and add it to the workbook's style table.
DataFormat createDataFormat()	It returns the instance of DataFormat for this workbook.
Font createFont()	It creates a new Font and add it to the workbook's font table.
Name createName()	It creates a new defined name in this workbook.
Sheet createSheet()	It creates a Sheet for this Workbook, adds it to the sheets and returns the high level representation.
Sheet createSheet(java.lang.String sheetname)	It creates a new sheet for this Workbook and return the high level representation.
void write(java.io.OutputStream stream) throws java.io.IOException	It writes out this workbook to an Outputstream.

Workbook, Sheet Row ,Cell Interfaces methods

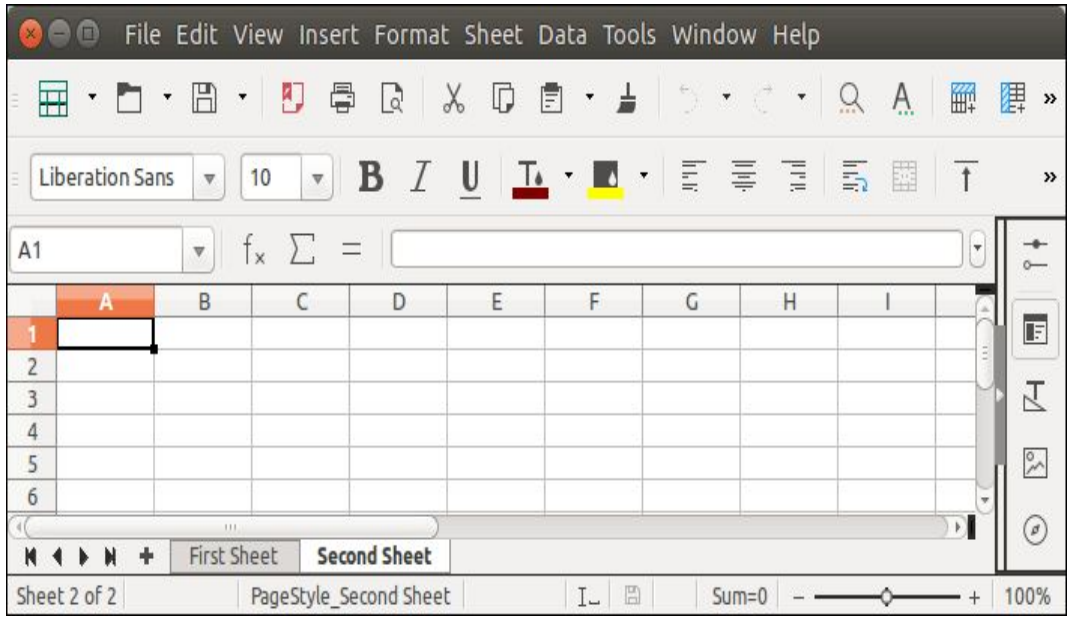
- To create a sheet in excel, POI provides a method **createSheet()** which creates a new sheet each time when function is called. While creating sheet consider the following points.
    1. Name must not exceed 31 characters.
    2. must not contain any of the any of the following characters (0x0000, 0x0003, colon (:), backslash (\), asterisk (*), question mark (?), forward slash (/), opening square bracket ([), closing square bracket (])).
- To create a cell in Excel Sheet, we can use Apache POI which provides **createCell()** method.

UNICAL ACADEMY

Example:

```
package poiexample;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.OutputStream;
import org.apache.poi.hssf.usermodel.HSSFWorkbook;
import org.apache.poi.ss.usermodel.Sheet;
import org.apache.poi.ss.usermodel.Workbook;
public class NewSheet {
   public static void main(String[] args) throws FileNotFoundException, IOException {
      Workbook wb = new HSSFWorkbook();
      try  (OutputStream fileOut = new FileOutputStream("Javatpoint.xls")) {
         Sheet sheet1 = wb.createSheet("First Sheet");
         Sheet sheet2 = wb.createSheet("Second Sheet");
         wb.write(fileOut);
      }catch(Exception e) {
         System.out.println(e.getMessage());
      }
   }
}
```

Output:



HSSFWorkbook, HSSFSheet, HSSFRow, HSSFCell methods

HSSFWorkbook :

- **This class has methods to read and write Microsoft Excel files in .xls format. It is compatible with MS-Office versions 97–2003.**

Methods and Constructors:

S.No.	Constructor & Description
1	**HSSFWorkbook()** Creates a new HSSFWorkbook object from scratch.
2	**HSSFWorkbook(DirectoryNode directory, boolean preserveNodes)** Creates a new HSSFWworkbook objectinside a specific directory.
3	**HSSFWorkbook(DirectoryNode directory, POIFSFileSystem fs, boolean preserveNodes)** Given a POIFSFileSystem object and a specific directory within it, it creates an SSFWorkbook object to read a specified workbook.
4	**HSSFWorkbook(java.io.InputStream s)** Creates a new HSSFWorkbook object using an input stream.
5	**HSSFWorkbook(java.io.InputStream s, boolean preserveNodes)** Constructs a POI file system around your input stream.
6	**HSSFWorkbook(POIFSFileSystem fs)** Constructs a new HSSFWorkbook object using a POIFSFileSystem object.
7	**HSSFWorkbook(POIFSFileSystem fs, boolean preserveNodes)** Given a POIFSFileSystem object, it creates a new HSSFWorkbook object to read a specified workbook.

Parameters used inside these constructors:

- **directory** − It is the POI filesystem directory to process from.
- **fs** − It is the POI filesystem that contains the workbook stream.
- **preservenodes** − This is an optional parameter that decides whether to preserve other nodes like macros. It consumes a lot of memory as it stores all the POIFileSystem in memory (if set).

HSSFSheet:

- It can create excel spreadsheets and it allows to format the sheet style and sheet data.

Class Constructors:

S. No.	Constructor & Description
1	**HSSFSheet(HSSFWorkbook workbook)** Creates new HSSFSheet called by HSSFWorkbook to create a sheet from scratch.
2	**HSSFSheet(HSSFWorkbook workbook, InternalSheet sheet)** Creates an HSSFSheet representing the given sheet obje

HSSFColor

Class Methods:

S. No.	Method & Description
1	**getIndex()** This method is used to get the index value of a nested class.

UNICAL ACADEMY

XSSFWorkbook, XSSFSheet, XSSFRow, XSSFCell methods

- It is a class that is used to represent both high and low level Excel file formats.

Class Methods:

S.No.	Method & Description
1	**createSheet()** Creates an XSSFSheet for this workbook, adds it to the sheets, and returns the high level representation.
2	**createSheet(java.lang.String sheetname)** Creates a new sheet for this Workbook and returns the high level representation.
3	**createFont()** Creates a new font and adds it to the workbook's font table.
4	**createCellStyle()** Creates a new XSSFCellStyle and adds it to the workbook's style table.
5	**createFont()** Creates a new font and adds it to the workbook's font table.
6	**setPrintArea(int sheetIndex, int startColumn, int endColumn, int startRow,int endRow)** Sets the print area of a given sheet as per the specified parameters.

UNICAL ACADEMY

XSSFSheet:

Class Methods:

S. No.	Method & Description
1	**addMergedRegion(CellRangeAddress region)** Adds a merged region of cells (hence those cells form one).
2	**autoSizeColumn(int column)** Adjusts the column width to fit the contents.
3	**iterator()** This method is an alias for rowIterator() to allow foreach loops
4	**addHyperlink(XSSFHyperlink hyperlink)** Registers a hyperlink in the collection of hyperlinks on this sheet

UNICAL ACADEMY

XSSFCell:

Class Methods:

S. No.	Method & Description
1	**setCellStyle(CellStyle style)** Sets the style for the cell.
2	**setCellType(int cellType)** Sets the type of cells (numeric, formula, or string).
3	**setCellValue(boolean value)** Sets a boolean value for the cell.
4	**setCellValue(java.util.Calendar value)** Sets a date value for the cell.
5	**setCellValue(double value)** Sets a numeric value for the cell.
6	**setCellValue(java.lang.String str)** Sets a string value for the cell.
7	**setHyperlink(Hyperlink hyperlink)** Assigns a hyperlink to this cell.

XSSFRow:

Class Methods:

S. No.	Method & Description
1	**createCell(int columnIndex)** Creates new cells within the row and returns it.
2	**setHeight(short height)** Sets the height in short units.

XSSFColor:

Class Methods:

S. No.	Method & Description
1	**setAuto(boolean auto)** Sets a boolean value to indicate that the ctColor is automatic and the system ctColor is dependent.
2	**setIndexed(int indexed)** Sets indexed ctColor value as system ctColor.

UNICAL ACADEMY

Read data from Excel file(XLS or XLSX)

- Save the written code in **Readsheet.java** file, and then compile and run it from the command prompt as follows −

```
$javac Readsheet.java
$java Readsheet
```

- If your system environment is configured with the POI library, it will compile and execute to generate the following output in the command prompt.

```
EMP ID        EMP NAME            DESIGNATION
tp01           Gopal               Technical Manager
tp02           Manisha             Proof Reader
tp03           Masthan             Technical Writer
tp04           Satish              Technical Writer
tp05           Krishna             Technical Writer
```

Write data to Excel file(XLS or XLSX)

- Here the employee data is given in a tabular form.

Emp Id	Emp Name	Designation
Tp01	Gopal	Technical Manager
TP02	Manisha	Proof Reader
Tp03	Masthan	Technical Writer
Tp04	Satish	Technical Writer
Tp05	Krishna	Technical Writer

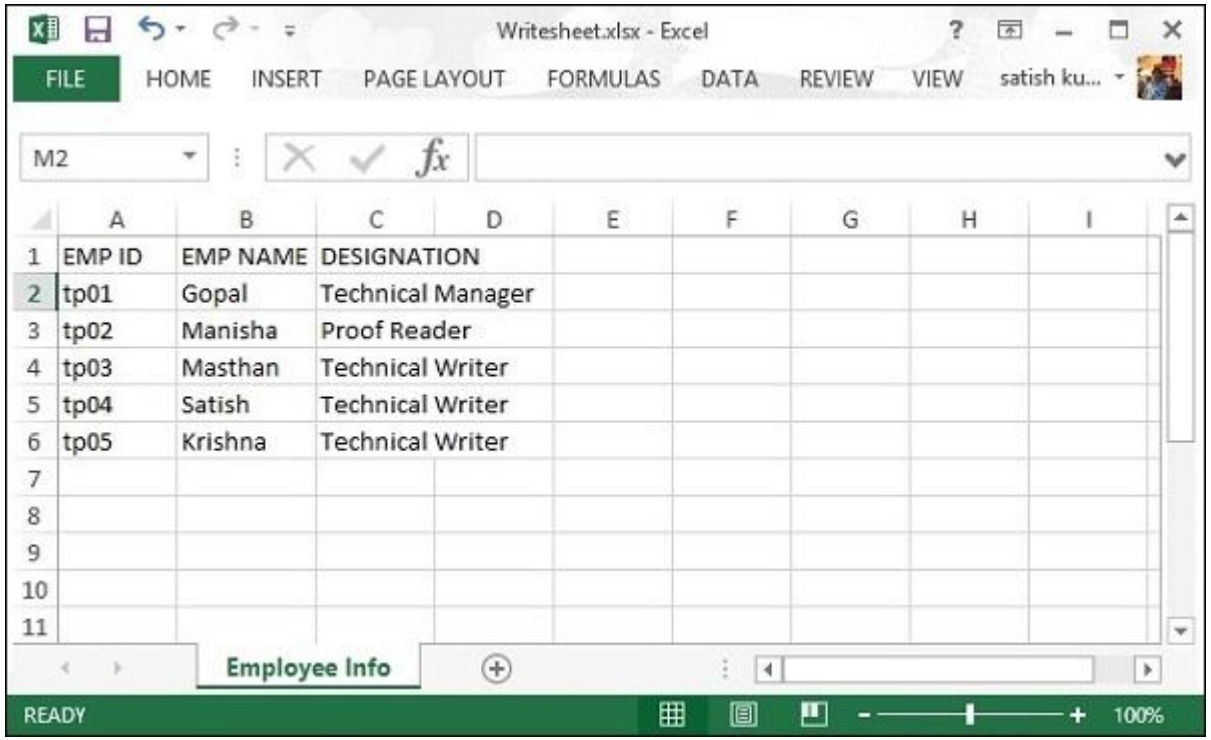
- Save the written Java code to write above data in to a Excel file as **Writesheet.java**, and then compile and run it from the command prompt as follows –

```
$javac Writesheet.java
$java Writesheet
```

- It will compile and execute to generate an Excel file named **Writesheet.xlsx** in your current directory and you will get the following output in the command prompt.

```
Writesheet.xlsx written successfully
```

The Writesheet.xlsx file looks as follows:



	A	B	C
1	EMP ID	EMP NAME	DESIGNATION
2	tp01	Gopal	Technical Manager
3	tp02	Manisha	Proof Reader
4	tp03	Masthan	Technical Writer
5	tp04	Satish	Technical Writer
6	tp05	Krishna	Technical Writer

UNICAL ACADEMY

Session Recap