DataDriven --> It is a framework

--> the operations to be performed is stored in an external file lie excel sheet

operation

read --> to read the the data from the excel

write --> to pass the data to the excel

Apache POI POI ; poor obfucation Implementation

🡪 an API that collecects java libraries and help us to read, write datas in excel.

Dependencies and Versions; should be add for the process(maven project also need)

POI -4.1.2

POI-OOXML - 4.1.2

POI-OOXML-schemas - 4.1.2

how to add maven project in it

maven repository --> poi 4.1.2 (copy and past the coading inbetween the dependency)

-->poi ooxml 4.1.2(copy and past the coading bellow the poi)

-->poi ooxml schemas 4.1.2(copy and past the coading bellow the poi ooml)

folder stracture ( workflow ) 🡪excel

🡪workbook --> material which is in excel

🡪sheet(en number of row and coloum(cell)) --> sheet - row -- coloum(cell)-- cell(cell value)

🡪workBook(Interface) ;(It get from java lib file)

🡪 A workbook representsa MicrosoftExcel file

🡪 used to creating and maintaining the spreadsheet

🡪 A workbook contains many sheets.

Types;

HSSF workbook(class)-- used to read / write data to microsoft Excel file in .xls formate.

MS - Office version 97-2003 XSSF-->Horrible spreadsheet formate

XSSF workbook (class) -- used to read / write data to microsoft Excel file in .xls or .xls formate.

MS - Office version 2007 or later XSSF-->XML spreadsheet formate

Syntax workbook w = new XSSFworkbook();

Sheet(Interface);

A sheet it contains number of rows and coloums

for

create sheet --> createsheet(name)

get the sheet --> getsheetAt(index should be pass)

getsheet(name should be pass in "\_")

Row(Interface);

A row represents a collection of cells, which is used to represent a row in sheet.

for

create Row -->createRow (index should be pass)

get the Row --> getRow (index should be pass )

get size(lenght) --> get Physical Number of Rows()

cell(Interface);

A cell is a indicated by a row and colum combination.

Data can be of the type such as string, numeric value or data.

for

create cell -->createcell(index should be pass)

get the cell --> getcell(index should be pass)

get size --> get Physical Number of cells()

get type of cell --> get cell type()

string - get string cell value()

return type is string

number - get numeric cell value() return type is double

the created exel sheet is coped and pased in ecplic

To read = .xlx convert.xlx

-->create a file --> create a object for file

-->than workbook is created (workbook w = xssfworkbook();) //ths is upcasting process

-->sheet(return type is webelement)-->by using getsheetAt travese into row

--> by using get row method we travese into cell

--> by using cellType method we can get the method

--> than IF conditions is used (refname.equals(Enum.STRING))

-->stringCellValue is used

-->numericCellValue is dome as the same method as row

to write  
--> create a excel sheet and copy and past in any location of folder  
--> create a file in ecplic  
-->create a object for it(fileInputstream fid = new fileInputstream(f);)    
-->creat a workbook (workbook w = xssfworkbook();) //ths is upcasting process  
-->create sheet (return type is webelement)  
-->createSheet for row(return type is create row)  
  -->create cell is created by row(return type is createcell)  
  -->createCell in it setcell value is created  
(wb.getSheet("credentials").getRow(0).createCell(1).setCellValue("password");  
  --> FileOutputStream fos = new FileOutputStream(f); --> object is created  
  -->wb.write(fos);  -->wb.close();   -->System.out.println("done");