**File Operations**

Task Description:

Write a Java program to write data to an Excel file using Apache POI library.

1. Create a new Excel workbook.

2. Create a new sheet with the name "Sheet1”.

| // Task 1: Create a new workbook and sheet  Workbook workbook = **new** XSSFWorkbook();  Sheet sheet = workbook.createSheet("Sheet1"); |
| --- |

3. Write the following data to the sheet:

Column headers: Name, Age, Email.

Row 1: John Doe, 30, john@test.com.

Row 2: Jane Doe, 28. john@test.com.

Row 3: Bob Smith, 35, jacky@example.com.

Row 4: Swapnil, 37, [swapnil@example.com](mailto:swapnil@example.com).

| // Task 2: Write data to the sheet  // Create header row  Row headerRow = sheet.createRow(0);  Cell headerCell0 = headerRow.createCell(0);  headerCell0.setCellValue("Name");  Cell headerCell1 = headerRow.createCell(1);  headerCell1.setCellValue("Age");  Cell headerCell2 = headerRow.createCell(2);  headerCell2.setCellValue("Email");  // Create data rows  String[][] data = { { "John Doe", "30", "john@test.com" }, { "Jane Doe", "28", "john@test.com" },  { "Bob Smith", "35", "jacky@example.com" }, { "Swapnil", "37", "swapnil@example.com" } };  **for** (**int** i = 0; i < data.length; i++) {  Row dataRow = sheet.createRow(i + 1);  **for** (**int** j = 0; j < data[i].length; j++) {  Cell dataCell = dataRow.createCell(j);  dataCell.setCellValue(data[i][j]);  }  } |
| --- |

4. Write a Java program to write data to an Excel file using Apache POI library.

| // Task 3 & 4: Write the workbook to an Excel file  FileOutputStream outputStream = **new** FileOutputStream("data.xlsx");  workbook.write(outputStream);  workbook.close();  outputStream.close();  System.***out***.println("Excel file created successfully!"); |
| --- |

5. Write a Java program to read data from an Excel file using Apache POI library

and print it in the console.

| // \*\* Task 5: Read data from the Excel file \*\*  // Open the Excel file  FileInputStream inputStream = **new** FileInputStream("data.xlsx");  // Create a workbook object from the file  Workbook readWorkbook = WorkbookFactory.*create*(inputStream);  // Get the first sheet (assuming you only have one)  Sheet readSheet = readWorkbook.getSheetAt(0);  // Iterate through rows and cells, printing data  **for** (**int** i = 0; i <= readSheet.getLastRowNum(); i++) {  Row row = readSheet.getRow(i);  // Skip header row  **if** (i == 0) {  **continue**;  }  **if** (row != **null**) {  **for** (**int** j = 0; j < row.getLastCellNum(); j++) {  Cell cell = row.getCell(j);  **if** (cell != **null**) {  **switch** (cell.getCellType()) {  **case** ***STRING***:  System.***out***.print(cell.getStringCellValue() + " ");  **break**;  **case** ***NUMERIC***:  System.***out***.print(cell.getNumericCellValue() + " ");  **break**;  **default**:  System.***out***.print(" (unsupported cell type) ");  }  } **else** {  System.***out***.print(" (empty cell) ");  }  }  System.***out***.println(); // New line after each row  }  }  // Close the input stream and workbook  inputStream.close();  readWorkbook.close();  }  } |
| --- |

***JAVA Code :-***

| **package** trainingtaskcompletion;  **import** java.io.FileInputStream;  **import** java.io.FileOutputStream;  **import** java.io.IOException;  **import** org.apache.poi.ss.usermodel.\*;  **import** org.apache.poi.xssf.usermodel.XSSFWorkbook;  **public** **class** ApachePOI {  **public** **static** **void** main(String[] args) **throws** IOException {  // Task 1: Create a new workbook and sheet  Workbook workbook = **new** XSSFWorkbook();  Sheet sheet = workbook.createSheet("Sheet1");  // Task 2: Write data to the sheet  // Create header row  Row headerRow = sheet.createRow(0);  Cell headerCell0 = headerRow.createCell(0);  headerCell0.setCellValue("Name");  Cell headerCell1 = headerRow.createCell(1);  headerCell1.setCellValue("Age");  Cell headerCell2 = headerRow.createCell(2);  headerCell2.setCellValue("Email");  // Create data rows  String[][] data = { { "John Doe", "30", "john@test.com" }, { "Jane Doe", "28", "john@test.com" },  { "Bob Smith", "35", "jacky@example.com" }, { "Swapnil", "37", "swapnil@example.com" } };  **for** (**int** i = 0; i < data.length; i++) {  Row dataRow = sheet.createRow(i + 1);  **for** (**int** j = 0; j < data[i].length; j++) {  Cell dataCell = dataRow.createCell(j);  dataCell.setCellValue(data[i][j]);  }  }  // Task 3 & 4: Write the workbook to an Excel file  FileOutputStream outputStream = **new** FileOutputStream("data.xlsx");  workbook.write(outputStream);  workbook.close();  outputStream.close();  System.***out***.println("Excel file created successfully!");  // \*\* Task 5: Read data from the Excel file \*\*  // Open the Excel file  FileInputStream inputStream = **new** FileInputStream("data.xlsx");  // Create a workbook object from the file  Workbook readWorkbook = WorkbookFactory.*create*(inputStream);  // Get the first sheet (assuming you only have one)  Sheet readSheet = readWorkbook.getSheetAt(0);  // Iterate through rows and cells, printing data  **for** (**int** i = 0; i <= readSheet.getLastRowNum(); i++) {  Row row = readSheet.getRow(i);  // Skip header row  **if** (i == 0) {  **continue**;  }  **if** (row != **null**) {  **for** (**int** j = 0; j < row.getLastCellNum(); j++) {  Cell cell = row.getCell(j);  **if** (cell != **null**) {  **switch** (cell.getCellType()) {  **case** ***STRING***:  System.***out***.print(cell.getStringCellValue() + " ");  **break**;  **case** ***NUMERIC***:  System.***out***.print(cell.getNumericCellValue() + " ");  **break**;  **default**:  System.***out***.print(" (unsupported cell type) ");  }  } **else** {  System.***out***.print(" (empty cell) ");  }  }  System.***out***.println(); // New line after each row  }  }  // Close the input stream and workbook  inputStream.close();  readWorkbook.close();  }  }  ***Output:-***  ERROR StatusLogger Log4j2 could not find a logging implementation. Please add log4j-core to the classpath. Using SimpleLogger to log to the console...  Excel file created successfully!  John Doe 30 john@test.com  Jane Doe 28 john@test.com  Bob Smith 35 jacky@example.com  Swapnil 37 swapnil@example.com |
| --- |