

```
eclipse-workspace - d7/src/main/java/athm/d7/D10P1.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer ×
> a1
> a2
> d7
D10P1.java × D10P2.java D10P3.java registerpage.java registerTest.java
athm.d7
D10P1
main(String[]): void
readData(XSSFSheet): void
writeData(XSSFSheet): void
1 package athm.d7;
2
3 import java.io.FileInputStream;
4 import java.io.FileOutputStream;
5 import java.io.IOException;
6 import org.apache.poi.ss.usermodel.Cell;
7 import org.apache.poi.ss.usermodel.CellType;
8 import org.apache.poi.ss.usermodel.Row;
9 import org.apache.poi.xssf.usermodel.XSSFSheet;
10 import org.apache.poi.xssf.usermodel.XSSFWorkbook;
11
12 public class D10P1
13 {
14     public static void main(String[] args) {
15         String excelFilePath = "D:\\data.xlsx";
16
17         try {
18             FileInputStream inputStream = new FileInputStream(excelFilePath);
19             XSSFWorkbook workbook = new XSSFWorkbook(inputStream);
20             XSSFSheet sheet = workbook.getSheetAt(0); // Assuming you want to work with the first sheet
21             readData(sheet);
22             writeData(sheet);
23             FileOutputStream outputStream = new FileOutputStream(excelFilePath);
24             workbook.write(outputStream);
25             workbook.close();
26             outputStream.close();
27
28             System.out.println("In console the data retrieved and Data written Successfully in excel!!");
29
30         } catch (IOException e) {
31             e.printStackTrace();
32         }
33     }
34     private static void readData(XSSFSheet sheet)
35     {
36         for (Row row : sheet) {
37             for (Cell cell : row) {
38                 CellType cellType = cell.getCellType();
39                 if (cellType == CellType.STRING) {
40                     System.out.print(cell.getStringCellValue() + " ");
41                 } else if (cellType == CellType.NUMERIC) {
42                     System.out.print(cell.getNumericCellValue() + " ");
43                 }
44             }
45             System.out.println();
46         }
47     }
48
49     private static void writeData(XSSFSheet sheet)
50     {
51         // Create a new row at the end of the excel and entering the data.
52         Row newRow = sheet.createRow(sheet.getLastRowNum() + 1);
53         // first create cell, then set cell value
54         Cell firstdata = newRow.createCell(0);
55         firstdata.setCellValue("First Data Written");
56         Cell seconddata = newRow.createCell(1);
57         seconddata.setCellValue(10);
58     }
59 }
60
```

```
eclipse-workspace - d7/src/main/java/athm/d7/D10P1.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer ×
> a1
> a2
> d7
D10P1.java × D10P2.java D10P3.java registerpage.java registerTest.java
athm.d7
D10P1
main(String[]): void
readData(XSSFSheet): void
writeData(XSSFSheet): void
19 XSSFWorkbook workbook = new XSSFWorkbook(inputStream);
20 XSSFSheet sheet = workbook.getSheetAt(0); // Assuming you want to work with the first sheet
21 readData(sheet);
22 writeData(sheet);
23 FileOutputStream outputStream = new FileOutputStream(excelFilePath);
24 workbook.write(outputStream);
25 workbook.close();
26 outputStream.close();
27
28 System.out.println("In console the data retrieved and Data written Successfully in excel!!");
29
30 } catch (IOException e) {
31     e.printStackTrace();
32 }
33
34 private static void readData(XSSFSheet sheet)
35 {
36     for (Row row : sheet) {
37         for (Cell cell : row) {
38             CellType cellType = cell.getCellType();
39             if (cellType == CellType.STRING) {
40                 System.out.print(cell.getStringCellValue() + " ");
41             } else if (cellType == CellType.NUMERIC) {
42                 System.out.print(cell.getNumericCellValue() + " ");
43             }
44         }
45         System.out.println();
46     }
47 }
48
49 private static void writeData(XSSFSheet sheet)
50 {
51     // Create a new row at the end of the excel and entering the data.
52     Row newRow = sheet.createRow(sheet.getLastRowNum() + 1);
53     // first create cell, then set cell value
54     Cell firstdata = newRow.createCell(0);
55     firstdata.setCellValue("First Data Written");
56     Cell seconddata = newRow.createCell(1);
57     seconddata.setCellValue(10);
58 }
59 }
60
```

```
eclipse-workspace - d7/src/main/java/athm/d7/D10P2.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer ×
  d7
  > a1
  > a2
  > d7
  > d7
    package athm.d7;
    import java.io.FileInputStream;
    import java.io.FileNotFoundException;
    import org.apache.poi.ss.usermodel.Row;
    import org.apache.poi.xssf.usermodel.XSSFSheet;
    import org.apache.poi.xssf.usermodel.XSSFWorkbook;
    import org.apache.poi.ss.usermodel.Cell;
    import org.apache.poi.ss.usermodel.CellType;
    public class D10P2
    {
    15 {
    16     public static void main(String[] args) throws Exception {
    17         // TODO Auto-generated method stub
    18         String excelFilePath = "D:\\data.xlsx";
    19
    20         try {
    21             //creating file input object
    22             FileInputStream inputStream=new FileInputStream(excelFilePath);
    23             //creating workbook object
    24             XSSFWorkbook workbook = new XSSFWorkbook(inputStream);
    25             //select sheet using object
    26             XSSFSheet sheet = workbook.getSheetAt(0);
    27
    28             //call readdata
    29             readdata(sheet);
    30
    31             //FileOutputStream outputStream=new FileOutputStream(excelFilePath);
    32             //workbook.write(outputStream);
    33             workbook.close();
    34             inputStream.close();
    35
    36         } catch (FileNotFoundException e) {
    37             // TODO Auto-generated catch block
    38             e.printStackTrace();
    39         }
    40     }
    41
    42     private static void readdata(XSSFSheet sheet)
    43
    44
    45
    46
    47
    48
    49
    50
    51
    52
    53
    54
    55
    56
    57
    58
    59
    Outline ×
    athm.d7
    > D10P2
    > main(String[]): void
    > readdata(XSSFSheet): vo
    Writable Smart Insert 1:1:0 21:45 06-06-2023
```

```
eclipse-workspace - d7/src/main/java/athm/d7/D10P2.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer ×
  d7
  > a1
  > a2
  > d7
  > d7
    String excelFilePath = "D:\\data.xlsx";
    try {
    21 //creating file input object
    22 FileInputStream inputStream=new FileInputStream(excelFilePath);
    23 //creating workbook object
    24 XSSFWorkbook workbook = new XSSFWorkbook(inputStream);
    25 //select sheet using object
    26 XSSFSheet sheet = workbook.getSheetAt(0);
    27
    28 //call readdata
    29 readdata(sheet);
    30
    31 //FileOutputStream outputStream=new FileOutputStream(excelFilePath);
    32 //workbook.write(outputStream);
    33 workbook.close();
    34 inputStream.close();
    35
    36 } catch (FileNotFoundException e) {
    37 // TODO Auto-generated catch block
    38 e.printStackTrace();
    39 }
    40
    41
    42 private static void readdata(XSSFSheet sheet)
    43 {
    44     for (Row row : sheet) { // for each row- moves row by row
    45         double s=0;
    46         for (Cell cell : row) { // moves each cell in the particular row
    47             CellType cellType = cell.getCellType();// returns kind of data in cell
    48             if (cellType == CellType.STRING) {
    49                 //System.out.print(cell.getStringCellValue() + "\t"); //get string values from cell
    50             } else if (cellType == CellType.NUMERIC) {
    51                 // System.out.print(cell.getNumericCellValue() + "\t");//get numerical values from cell
    52                 s+=cell.getNumericCellValue();
    53             }
    54             System.out.println(s);
    55         }
    56     }
    57 }
    58
    59
    Outline ×
    athm.d7
    > D10P2
    > main(String[]): void
    > readdata(XSSFSheet): vo
    Writable Smart Insert 1:1:0 21:45 06-06-2023
```

```
eclipse-workspace - d7/src/main/java/athm/d7/D10P3.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer ×
D10P1.java D10P2.java D10P3.java × registerpage.java registertest.java
D7 > src/main/java > athm.d7 > D10P3 > readdata(XSSFSheet):void
Outline ×
athm.d7
D10P3
main(String[]):void
readdata(XSSFSheet):vo

1 package athm.d7;
2
3 import java.io.FileInputStream;
4 import java.io.FileNotFoundException;
5 import java.util.ArrayList;
6 import java.util.Collections;
7
8 import org.apache.poi.ss.usermodel.Cell;
9 import org.apache.poi.ss.usermodel.CellType;
10 import org.apache.poi.ss.usermodel.Row;
11 import org.apache.poi.xssf.usermodel.XSSFSheet;
12 import org.apache.poi.xssf.usermodel.XSSFWorkbook;
13
14 public class D10P3
15 {
16     public static void main(String[] args) throws Exception {
17
18         // TODO Auto-generated method stub
19         String excelFilePath = "D:\\data.xlsx";
20         try {
21             //creating file input object
22             FileInputStream inputStream=new FileInputStream(excelFilePath);
23             //creating workbook object
24             XSSFWorkbook workbook = new XSSFWorkbook(inputStream);
25             //select sheet using object
26             XSSFSheet sheet = workbook.getSheetAt(0);
27             //call readdata
28             readdata(sheet);
29             workbook.close();
30             inputStream.close();
31         } catch (FileNotFoundException e) {
32             // TODO Auto-generated catch block
33             e.printStackTrace();
34         }
35     }
36
37     private static void readdata(XSSFSheet sheet)
38     {
39         for (Row row : sheet) { // for each row- moves row by row
40             ArrayList<Double> arrayList=new ArrayList<Double>();
41
42             for (Cell cell : row) { // moves each cell in the particular row
```

```
Writeable Smart Insert 51:21:1886 21:46 06-06-2023
Type here to search
eclipse-workspace - d7/src/main/java/athm/d7/D10P3.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer ×
D10P1.java D10P2.java D10P3.java × registerpage.java registertest.java
D7 > src/main/java > athm.d7 > D10P3 > readdata(XSSFSheet):void
Outline ×
athm.d7
D10P3
main(String[]):void
readdata(XSSFSheet):vo

15 {
16     public static void main(String[] args) throws Exception {
17
18         // TODO Auto-generated method stub
19         String excelFilePath = "D:\\data.xlsx";
20         try {
21             //creating file input object
22             FileInputStream inputStream=new FileInputStream(excelFilePath);
23             //creating workbook object
24             XSSFWorkbook workbook = new XSSFWorkbook(inputStream);
25             //select sheet using object
26             XSSFSheet sheet = workbook.getSheetAt(0);
27             //call readdata
28             readdata(sheet);
29             workbook.close();
30             inputStream.close();
31         } catch (FileNotFoundException e) {
32             // TODO Auto-generated catch block
33             e.printStackTrace();
34         }
35     }
36
37     private static void readdata(XSSFSheet sheet)
38     {
39         for (Row row : sheet) { // for each row- moves row by row
40             ArrayList<Double> arrayList=new ArrayList<Double>();
41
42             for (Cell cell : row) { // moves each cell in the particular row
43                 CellType cellType = cell.getCellType();// returns kind of data in cell
44                 if (cellType == CellType.STRING) {
45                     //System.out.print(cell.getStringCellValue() + "\t"); //get string values from cell
46                 } else if (cellType == CellType.NUMERIC) {
47                     // System.out.print(cell.getNumericCellValue() + "\t");//get numerical values from cell
48                     arrayList.add(cell.getNumericCellValue());
49                 }
50             }
51             System.out.println(Collections.min(arrayList));
52         }
53     }
54 }
55 }
56 }
```

