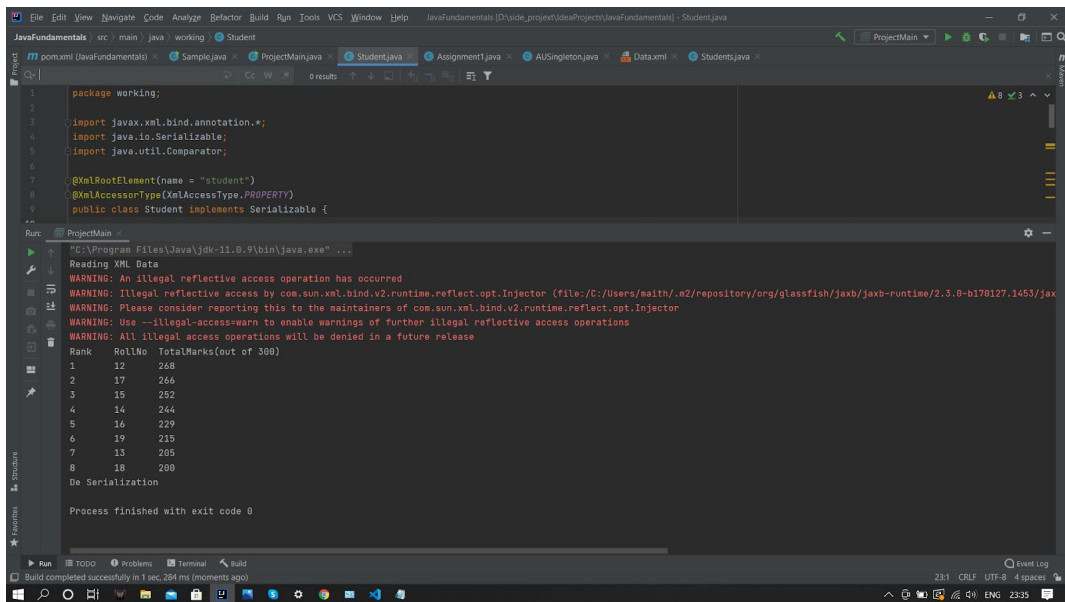


JAVA FUNDAMENTALS ASSIGNMENT

Maithreyan K

1) Read an XML file “student.xml” containing list of student data in the following format, deserialize them into java objects, then serialize the unique <rollnumber, total marks(Phy+chem+math)> to a text file “student.txt” with appropriate exception handling.



```
package working;

import javax.xml.bind.annotation.*;
import java.io.Serializable;
import java.util.Comparator;

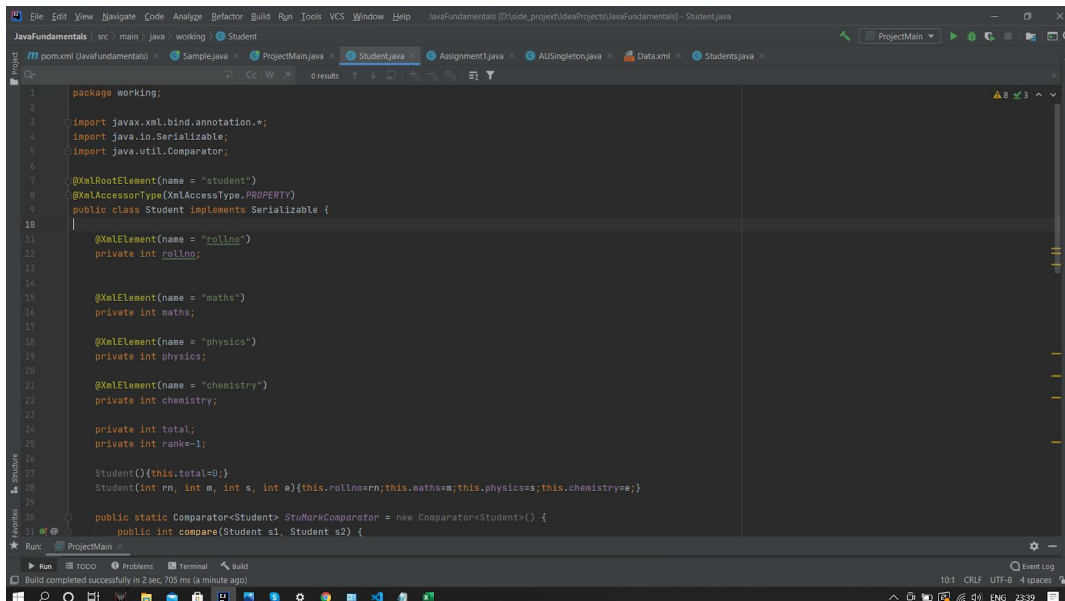
@XmlRootElement(name = "student")
@XmlAccessorType(XmlAccessType.PROPERTY)
public class Student implements Serializable {

    Run: ProjectMain
    "C:\Program Files\Java\jdk-11.0.9\bin\java.exe" ...
    Reading XML Data
    WARNING: An illegal reflective access operation has occurred
    WARNING: Illegal reflective access by com.sun.xml.bind.v2.runtime.reflect.opt.Injector (file:/C:/Users/maith/.m2/repository/org/glassfish/jaxb/jaxb-runtime/2.3.0-b170127.1453/jaxb-runtime-2.3.0-b170127.1453.jar) to method java.lang.ProcessImpl.getErrorStream()Ljava/io/InputStream;
    WARNING: Please consider reporting this to the maintainers of com.sun.xml.bind.v2.runtime.reflect.opt.Injector
    WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
    WARNING: All illegal access operations will be denied in a future release

    Rank RollNo TotalMarks(out of 300)
    1 12 268
    2 17 266
    3 15 292
    4 14 264
    5 16 229
    6 19 215
    7 13 205
    8 18 200

    De Serialization

    Process finished with exit code 0
```



```
package working;

import javax.xml.bind.annotation.*;
import java.io.Serializable;
import java.util.Comparator;

@XmlRootElement(name = "student")
@XmlAccessorType(XmlAccessType.PROPERTY)
public class Student implements Serializable {

    @XmlElement(name = "rollno")
    private int rollno;

    @XmlElement(name = "maths")
    private int maths;

    @XmlElement(name = "physics")
    private int physics;

    @XmlElement(name = "chemistry")
    private int chemistry;

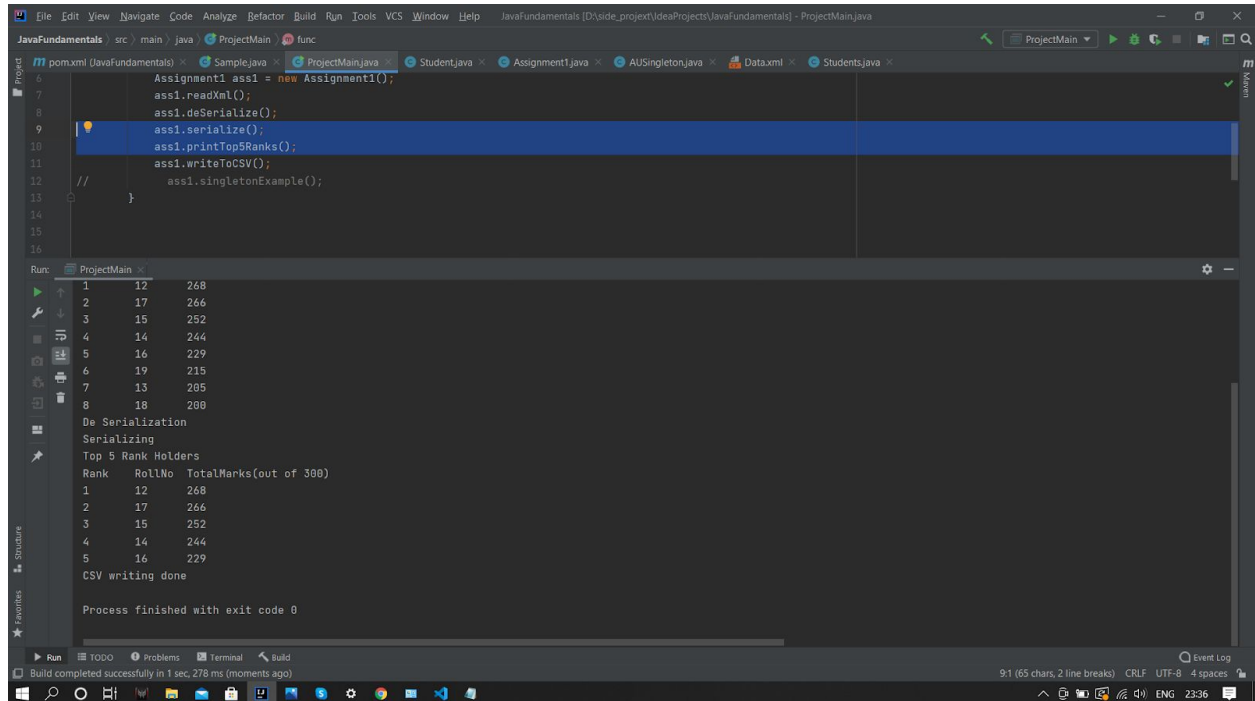
    private int total;
    private int rank=1;

    Student(){this.total=0;}
    Student(int rn, int m, int s, int e){this.rollno=rn;this.maths=m;this.physics=s;this.chemistry=e;}

    public static Comparator<Student> StuMarkComparator = new Comparator<Student>() {
        public int compare(Student s1, Student s2) {
```

2. Secondly, deserialize the previously stored student.txt file with roll num and total marks.

Then, serialize the top 5 students' roll numbers and total marks into excel or csv file in the following format.

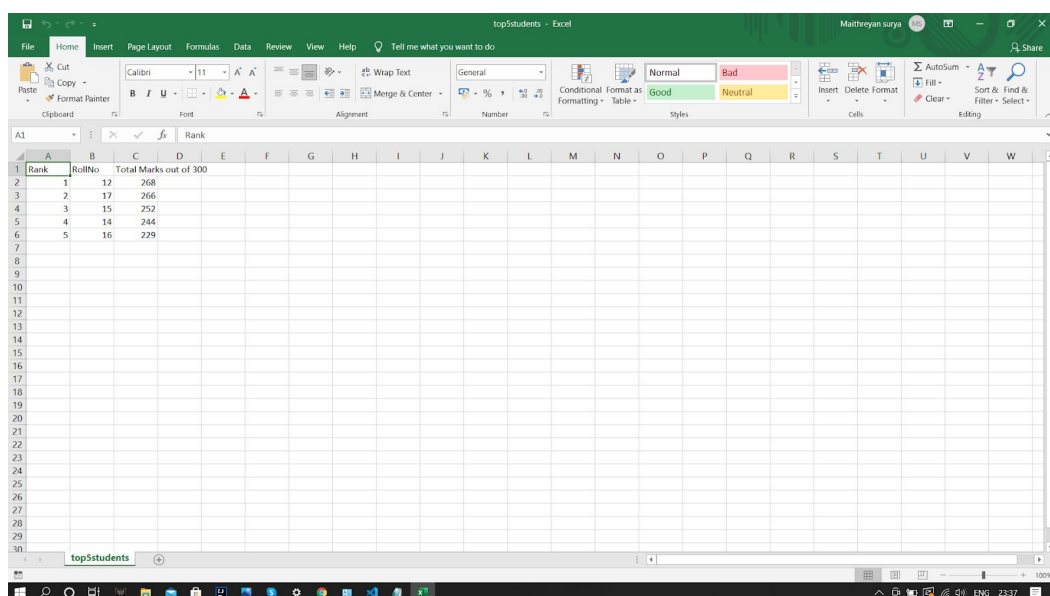


The screenshot shows an IDE with a Java project. The code in `ProjectMain.java` is as follows:

```
6 Assignment1 ass1 = new Assignment1();
7 ass1.readXml();
8 ass1.deserialize();
9 ass1.serialize();
10 ass1.printTopSRanks();
11 ass1.writeToCSV();
12 // ass1.singletonExample();
13 }
14
15
16
```

The Run window shows the following output:

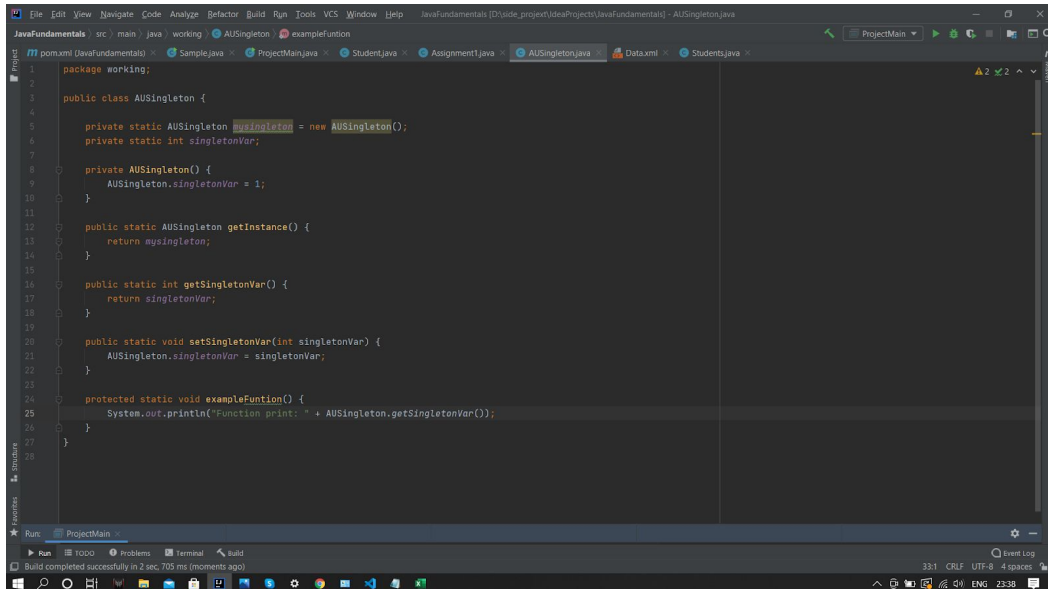
```
1 12 268
2 17 266
3 15 252
4 14 244
5 16 229
6 19 215
7 13 205
8 18 200
De Serialization
Serializing
Top 5 Rank Holders
Rank RollNo TotalMarks(out of 300)
1 12 268
2 17 266
3 15 252
4 14 244
5 16 229
CSV writing done
Process finished with exit code 0
```



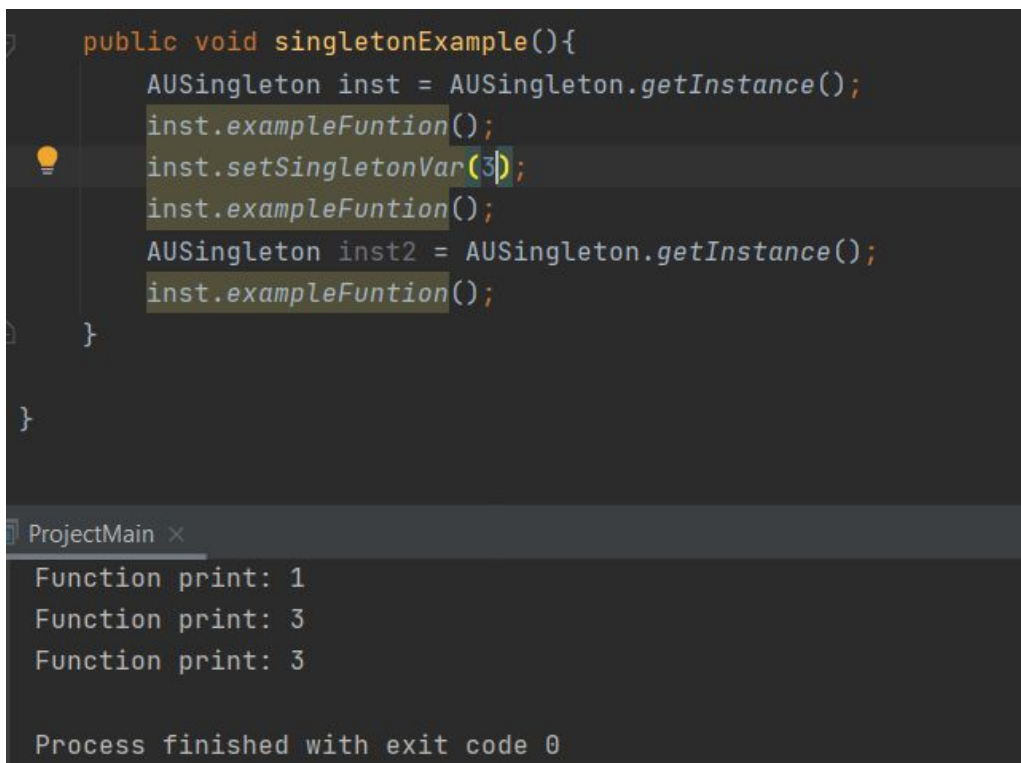
The screenshot shows an Excel spreadsheet titled "top5Students - Excel". The data is as follows:

Rank	RollNo	Total Marks out of 300
1	12	268
2	17	266
3	15	252
4	14	244
5	16	229

3. Write the implementation for singleton class.



```
1 package working;
2
3 public class AUSingleton {
4
5     private static AUSingleton mySingleton = new AUSingleton();
6     private static int singletonVar;
7
8     private AUSingleton() {
9         AUSingleton.singletonVar = 1;
10    }
11
12    public static AUSingleton getInstance() {
13        return mySingleton;
14    }
15
16    public static int getSingletonVar() {
17        return singletonVar;
18    }
19
20    public static void setSingletonVar(int singletonVar) {
21        AUSingleton.singletonVar = singletonVar;
22    }
23
24    protected static void exampleFunction() {
25        System.out.println("Function print: " + AUSingleton.getSingletonVar());
26    }
27
28 }
```



```
public void singletonExample(){
    AUSingleton inst = AUSingleton.getInstance();
    inst.exampleFuntion();
    inst.setSingletonVar(3);
    inst.exampleFuntion();
    AUSingleton inst2 = AUSingleton.getInstance();
    inst.exampleFuntion();
}

}
```

ProjectMain x

```
Function print: 1
Function print: 3
Function print: 3

Process finished with exit code 0
```

