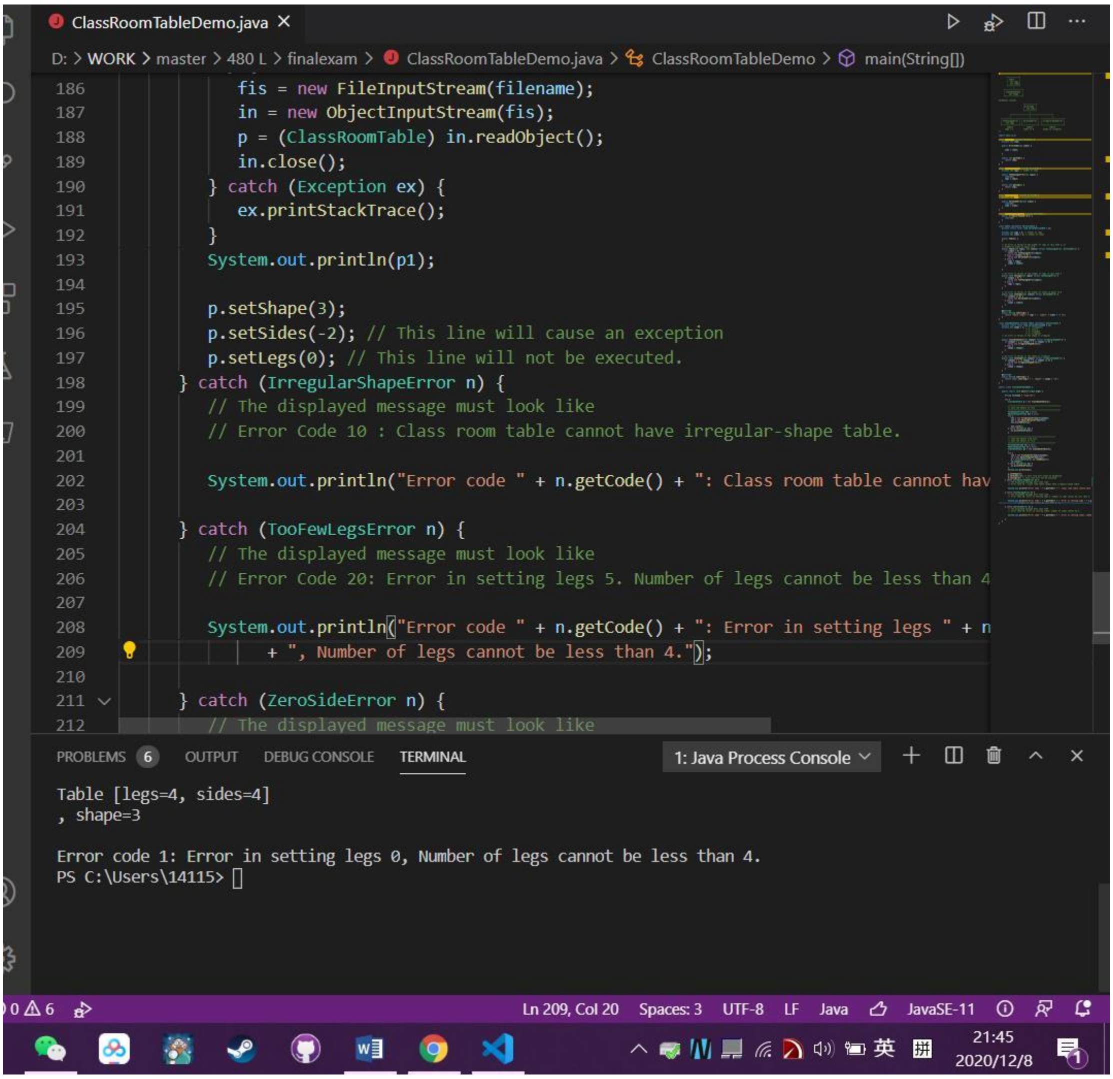
```
/*

             +-----------+
             | Table        |
             |    int legs |
             |    int sides|
             +-----------+
                       |
        +-----------------+
        | ClassRoomTable    |
        |       int shape       |
        +-----------------+

Exception classes

                                   +-----------+
                                   |ErrorCode     |
                                   |    int code |
                                   +-----------+
                                             |
                   +------------------+---------------------+
                   |                              |                                  |
    +-----------------+ +---------------+    +---------------------+
    | TooFewLegsError | | ZeroSideError |    | IrregularShapeError |
    |       int legs         | |                         |    |                                    |
    +-----------------+ +---------------+    +---------------------+
             code=1                        code=2                           code=3
          legs < 4                  sides == 0              shape is irregular
*/

import java.io.*;

class ErrorCode extends Exception {
     private int code;

     public ErrorCode(int code1) {

          code = code1;

     }

     public int getCode() {
          return code;
     }
```

```
}

class TooFewLegsError extends ErrorCode {
    private int legs; // number of legs

    public TooFewLegsError(int legs1) {
        super(1);
        legs = legs1;
    }

    public int getLegs() {
        return legs;
    }
}

class ZeroSideError extends ErrorCode {
    private int side;
      // public ZeroSideError() {
      //          super(2);}

    public ZeroSideError(int side1) {
        super(2);
        side = side1;
    }
}

class IrregularShapeError extends ErrorCode {
    public IrregularShapeError() {
        super(3);
    }
}

class Table implements Serializable {
    private static final long serialVersionUID = 1L;

    private int legs = 4; // number of legs
    private int sides = 4; // number of sides

    public Table() {
    }

    // An error is thrown if the number of legs is less than 4, or
    // number of sides is equal to 0
    public Table(int legs1, int sides1) throws TooFewLegsError, ZeroSideError {
        if (legs1 < 4) {
            throw new TooFewLegsError(legs1);
        } else if (sides1 == 0) {
            throw new ZeroSideError(sides1);
        } else {
            legs = legs1;
            sides = sides1;
        }

    }

    // An error is thrown if the number of legs is less than 4
    public void setLegs(int legs1) throws TooFewLegsError {
        if (legs1 < 4) {
            throw new TooFewLegsError(legs1);
        } else {
            legs = legs1;
        }
    }

    // An error is thrown if the number of sides is equal to 0
    public void setSides(int sides1) throws ZeroSideError {
        if (sides1 == 0) {
            throw new ZeroSideError(sides1);
        } else {
            sides = sides1;
        }
    }

    @Override
    public String toString() {
        return "Table [legs=" + legs + ", sides=" + sides + "] \n";
    }
}

class ClassRoomTable extends Table implements Serializable {
    private static final long serialVersionUID = 1L;
    private int shape = 1; // 1: rectangular
                                  // 2: circle
                                  // 3: triangle
                                  // 4: irregular
    // An error is thrown if the shape is irregular

    public ClassRoomTable(int shape1) throws IrregularShapeError {
        if (shape1 < 1 || shape1 > 3 || shape1 == 4) {
            throw new IrregularShapeError();
        } else {
            shape = shape1;
        }
    }

    // An error is thrown if the shape is irregular
    public void setShape(int shape1) throws IrregularShapeError {
        if (shape1 < 1 || shape1 > 3 || shape1 == 4) {
            throw new IrregularShapeError();
        } else {
            shape = shape1;
        }
    }

    @Override
    public String toString() {
        return super.toString() + ", shape=" + shape + "\n";
    }
}

public class ClassRoomTableDemo {

    public static void main(String[] args) {

        String filename = "time.ser";

        try {
            ClassRoomTable p = new ClassRoomTable(2);

            //////////////////////////////////////////////////
            // save the object to file
            //////////////////////////////////////////////////
            FileOutputStream fos = null;
            ObjectOutputStream out = null;
            try {
                fos = new FileOutputStream(filename);
                out = new ObjectOutputStream(fos);
                out.writeObject(p);

                out.close();
            } catch (Exception ex) {
                ex.printStackTrace();
            }

            ////////////////////////////////////////////
            // read the object from file
            // save the object to file
            ////////////////////////////////////////////
            FileInputStream fis = null;
            ObjectInputStream in = null;
            ClassRoomTable p1 = new ClassRoomTable(3);
            ;
            try {
                fis = new FileInputStream(filename);
                in = new ObjectInputStream(fis);
                p1 = (ClassRoomTable) in.readObject();
                in.close();
            } catch (Exception ex) {
                ex.printStackTrace();
            }
            System.out.println(p1);

            p.setShape(3);
            p.setSides(-2); // This line will cause an exception
            p.setLegs(0); // This line will not be executed.
        } catch (IrregularShapeError n) {
            // The displayed message must look like
            // Error Code 10 : Class room table cannot have irregular-shape table.

            System.out.println("Error code " + n.getCode() + ": Class room table cannot have
irregular-shape table.");

        } catch (TooFewLegsError n) {
            // The displayed message must look like
            // Error Code 20: Error in setting legs 5. Number of legs cannot be less than 4.

            System.out.println("Error code " + n.getCode() + ": Error in setting legs " + n.getLegs()
                    + ", Number of legs cannot be less than 4.");

        } catch (ZeroSideError n) {
            // The displayed message must look like
            // Error Code 30: Error in setting sides, number of sides cannot be 0

            System.out.println("Error code " + n.getCode() + ": Error in setting sides, number of
sides cannot be 0");
```

```
            }
        }
    }
```

THE OUTPUT SCREENSHOOT:



```
            fis = new FileInputStream(filename);
            in = new ObjectInputStream(fis);
            p = (ClassRoomTable) in.readObject();
            in.close();
         } catch (Exception ex) {
            ex.printStackTrace();
         }
         System.out.println(p1);

         p.setShape(3);
         p.setSides(-2); // This line will cause an exception
         p.setLegs(0); // This line will not be executed.
      } catch (IrregularShapeError n) {
         // The displayed message must look like
         // Error Code 10 : Class room table cannot have irregular-shape table.

         System.out.println("Error code " + n.getCode() + ": Class room table cannot hav

      } catch (TooFewLegsError n) {
         // The displayed message must look like
         // Error Code 20: Error in setting legs 5. Number of legs cannot be less than 4

         System.out.println("Error code " + n.getCode() + ": Error in setting legs " + n
               + ", Number of legs cannot be less than 4.");

      } catch (ZeroSideError n) {
         // The displayed message must look like
```

```
Table [legs=4, sides=4]
, shape=3

Error code 1: Error in setting legs 0, Number of legs cannot be less than 4.
PS C:\Users\14115> 
```