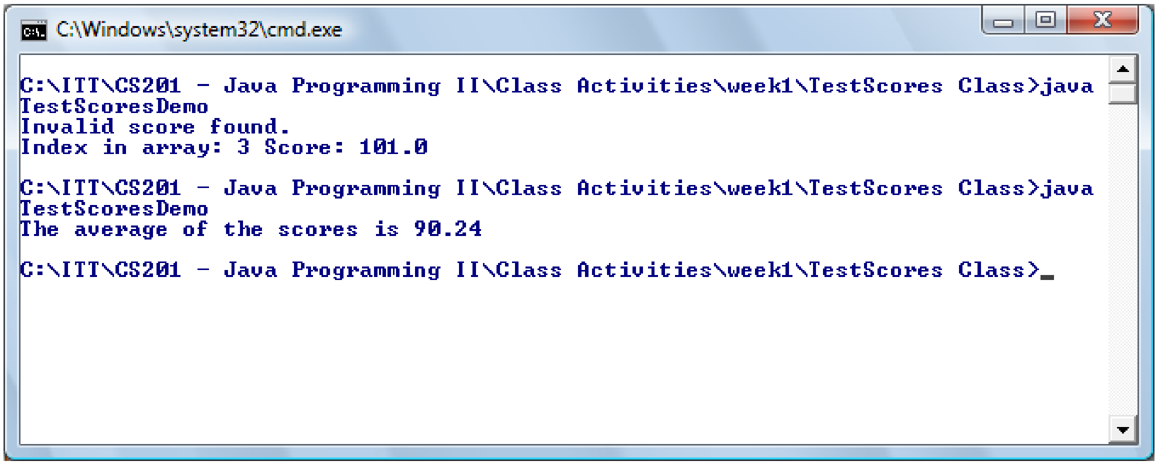
Labs

Lab 1.1: TestScores Class

What is the purpose?

In this lab, you will write a class named TestScores. The class constructor should accept an array of test scores as its argument. The class should have a method that returns the average of the test scores. If any test score in the array is negative or greater than 100, the class should have an IllegalArgumentException.

See Figure 1-1-1 for a sample of output:



```
C:\ITT\CS201 - Java Programming II\Class Activities\week1\TestScores Class>java
TestScoresDemo
Invalid score found.
Index in array: 3 Score: 101.0

C:\ITT\CS201 - Java Programming II\Class Activities\week1\TestScores Class>java
TestScoresDemo
The average of the scores is 90.24

C:\ITT\CS201 - Java Programming II\Class Activities\week1\TestScores Class>_
```

Figure 1-1-1

What are the steps?

- **Task 1**

  Procedure:

  1. Create a Java file called TestScores.java.
  2. Create a constructor that initializes an object with an array of scores. If the array contains an invalid value (less than 0 or greater than 100), an IllegalArgumentException is thrown.
  3. Create a method called getAverage that returns the average of the object's test scores.
     Figure 1-1-2 represents a program skeleton:

```
/**
 TestScores Class
*/

public class TestScores
{
 // Variable to reference an array holding test scores
 private double[] scores;
 /**
  The constructor initializes an object with an array of scores.
  If the array contains an invalid value (less than 0 or greater
  than 100) an exception is thrown.
  @param s The array of test scores.
  @exception IllegalArgumentException When the
  argument array contains an invalid value.
 */

 public TestScores(double[] s) throws IllegalArgumentException
 {
  // Create an array to hold the scores passed
  // as an argument.
  scores = new double[s.length];

  // Copy the scores passed as an argument into
  // the new array. Check for illegal values as
  // they are copied.
 }

 /**
  The getAverage method returns the average
  of the object's test scores.
  @return The average of the object's test scores.
 */

 public double getAverage()
 {
  double total = 0.0; // Accumulator

  // Accumulate the sum of the scores.


  // return the average.

 }
}
```

Figure 1-1-2: Sample Program Skeleton 1

4. Create a demo program to implement the TestScores class. Figure 1-1-3 is a program skeleton:

```
/**
   Demo program for the TestScores Class
*/

public class TestScoresDemo
{
   public static void main(String[] args)
   {
      // An array with test scores.
      // Notice that element 3 contains an invalid score.
      Double[] badScores = {97.5, 66.7, 88.0, 101.0, 99.0 };

      // Another array with test scores.
      // All of these scores are good.
      Double[] goodScores = {97.5, 66.7, 88.0, 100.0, 99.0 };

      try
      {
         // Create a TestScores object initialized with badScores.

         // Create a TestScores object initialized with goodScores.

         // The following statement should not execute.

      }
      catch (IllegalArgumentException e)
      {
         System.out.println(“Invalid score found.\n” +
e.getMessage());
      }
   }
}
```

Figure 1-1-3: Sample Program Skeleton 2

5. Submit your Java code and sample output to your instructor.

Did it work?

Were you able to:

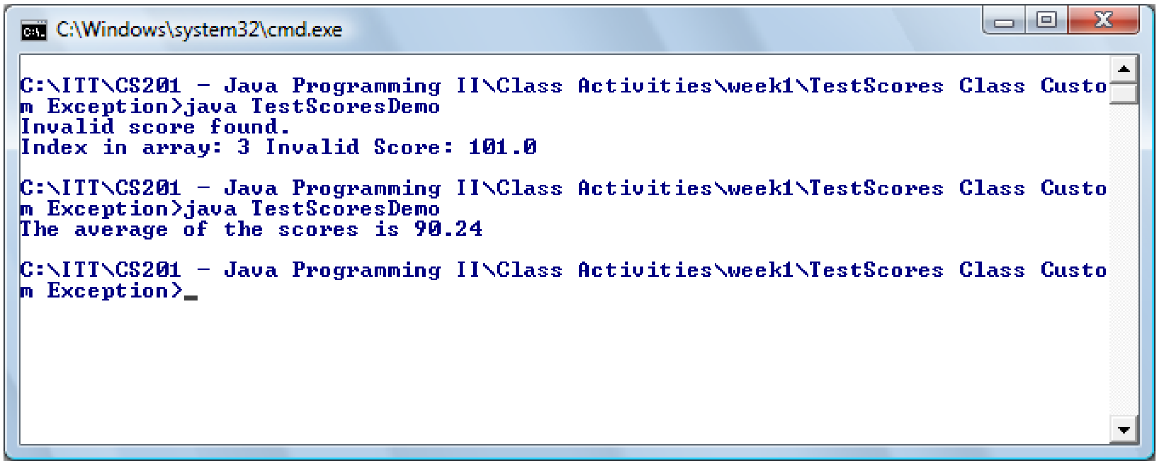
- Initialize an array of scores with a constructor?
- Throw an IllegalArgumentException when the array encounters some invalid values?
- Return the average of the array by the getAverage() method?

Lab 1.2: TestScores Class Custom Exception

What is the purpose?

In this lab, you will write an exception class named InvalidTestScore. Modify the TestScores class you wrote in Lab 1.1 so that it throws an InvalidTestScore exception if any test score in the array is invalid.

Figure 1-2-1 shows a sample of output:



```
C:\ITT\CS201 - Java Programming II\Class Activities\week1\TestScores Class Custo
m Exception>java TestScoresDemo
Invalid score found.
Index in array: 3 Invalid Score: 101.0

C:\ITT\CS201 - Java Programming II\Class Activities\week1\TestScores Class Custo
m Exception>java TestScoresDemo
The average of the scores is 90.24

C:\ITT\CS201 - Java Programming II\Class Activities\week1\TestScores Class Custo
m Exception>_
```

Figure 1-2-1

What are the steps?

- **Task 1**

Procedure:

1. Create a Java file called TestScores.java.
2. Create a constructor that initializes an object with an array of scores. If the array contains an invalid value (less than 0 or greater than 100), a custom-made exception called InvalidTestScore is thrown.
3. Create a method called getAverage that returns the average of the object's test scores.

   Figure 1-2-2 is a program skeleton:

```
/**
   TestScores Class
*/

public class TestScores
{
   // Variable to reference an array holding test scores
   private double[] scores;

   /**
      The constructor initializes an object with an array of scores.
      If the array contains an invalid value (less than 0 or greater
      than 100) an exception is thrown.
      @param s The array of test scores.
      @exception InvalidTestScore When the argument
       array contains an invalid value.
   */

   public TestScores(double[] s) throws InvalidTestScore
   {
      // Create an array to hold the scores passed
      // as an argument.
      scores = new double[s.length];

      // Copy the scores passed as an argument into the new array.
      // Check for illegal values as they are copied.
   }

   /**
      The getAverage method returns the average of the object's test
scores.
      @return The average of the object's test scores.
   */

   public double getAverage()
   {
      double total = 0.0;  // Accumulator

      // Accumulate the sum of the scores.


      // return the average.
      return (total / scores.length);
   }
}
```

Figure 1-2-2: Sample Program Skeleton 1

4. Create a class called InvalidTestScore that extends Exception class. This class throws exceptions by TestScore objects when an invalid test score is passed into the constructor. The constructor reports the index number of the array containing the invalid score and the value stored in that index.
   Figure 1-2-3 is a program skeleton:

```
/**
 InvalidTestScore exceptions are thrown by
 TestScore objects when an invalid test score is
 passed into the constructor.
*/

public class InvalidTestScore extends Exception
{
 /**
  No-arg constructor
 */

 public InvalidTestScore()
 {

 }

 /**
  This constructor reports the element number of the array
containing
  the invalid score and the value stored in that element.
  @param index The index of array containing the invalid score.
  @param score The invalid score.
 */
 public InvalidTestScore(int index, double score)
 {

 }
}
```

Figure 1-2-3: Sample Program Skeleton 2

5. Create a demo program to implement the TestScores class. Figure 1-2-4 shows a sample. Print your output.

```
/**
   Demo program for the TestScores Class
*/

public class TestScoresDemo
{
   public static void main(String[] args)
   {
      // An array with test scores.
      // Notice that element 3 contains an invalid score.
      double[] badScores = {97.5, 66.7, 88.0, 101.0, 99.0 };

      // Another array with test scores.
      // All of these scores are good.
      double[] goodScores = {97.5, 66.7, 88.0, 100.0, 99.0 };

      try
      {
         // Create a TestScores object initialized with badScores.

         // Create a TestScores object initialized with goodScores.

         // The following statement should not execute.

      }
      catch (InvalidTestScore e)
      {
         System.out.println("Invalid score found.\n" +
e.getMessage());
      }
   }
}
```

Figure 1-2-4: Sample Demo Program

6. Submit your Java code and sample output to your instructor.

Did it work?

Were you able to:

- Initialize an array of scores with a constructor?
- Throw an InvalidTestScore when the array encounters some invalid values?
- Return the average of the array by the getAverage() method?