```
/*
      Class to produce a transaction log based on the data
      entered for individual customers
*/
import java.io.*;
import java.text.NumberFormat;

public class DME6ATransactionLog
{
  private String customerNumber, readingMonth, readingDay, readingYear,
      beginReading, endReading;

  // Stream to write file
  public static FileOutputStream fout;

  public DME6ATransactionLog()
  {
      customerNumber = "00000";
      readingMonth = "00";
      readingDay = "00";
      readingYear = "0000";
      beginReading = "00000";
      endReading = "00000";

      openTransLog();
   }

   public static void openTransLog()
   {
     try
     {
     // Open an output stream
       fout = new FileOutputStream ("TransactionLog.txt");
     }
     // Catches any error conditions
     catch (FileNotFoundException e)
     {
       System.err.println ("Unable to open file. "+ e);
       System.exit(-1);
     }
   }
```

```
public static void closeTransLog()
{
  try
  {
    // Close our output stream
      fout.close();
  }
  // Catches any error conditions
  catch (IOException e)
  {
    System.err.println ("Unable to close file. "+ e);
    System.exit(-1);
  }
}

public static void writeTransRecord(String custNo, String readDay,
         String readMonth, String readYear, String beginRead, String endRead)
{
  String transRecord = "";
  int itemLength, beginPoint = 0;
  transRecord = transRecord + FixField(custNo, 5);
  transRecord = transRecord + FixField(readDay, 2);
  transRecord = transRecord + FixField(readMonth, 2);
  transRecord = transRecord + FixField(readYear, 4);
  transRecord = transRecord + FixField(beginRead, 5);
  transRecord = transRecord + FixField(endRead, 5);
  try
  {
  // Print a line of text
    if (transRecord.length() > 0)
    {
      new PrintStream(fout).println(transRecord);
    }
    else
      throw new IOException();
  }
  // Catches any error conditions
  catch (IOException e)
  {
    System.err.println ("Unable to write record to file");
    System.exit(-1);
  }
```

```
    }
    public static String FixField(String stringField, int stringLength)
    {
      String resultField = "";
      int itemLength, beginPoint = 0;
      resultField = stringField;
      resultField = "000000" + resultField;
      itemLength = resultField.length();
      beginPoint = itemLength - stringLength;
      resultField = resultField.substring(beginPoint, itemLength);
      return resultField;
    }
}
```