/\*\*

Time.java

Represents time in hours and minutes using

the customary conventions.

\*/

public class Time

{

private int hours; // Conventional hours

private int minutes; // Conventional minutes

private boolean afternoon; // Flag for afternoon

/\*\*

Constructs a customary time (12 hours, am or pm)

from a military time ##:##

@param militaryTime Time in the military

format ##:##

\*/

public Time(String militaryTime)

{

// Check to make sure something was entered

if (militaryTime == null)

{

System.out.println(militaryTime +

" is not a " +

"valid miliary time." );

}

// Check to make sure there are 5 characters

else if ( militaryTime.length() != 5) // CONDITION TO CHECK LENGTH OF STRING)

{

System.out.println(militaryTime +

" is not a " +

"valid miliary time." );

}

else

{

// Check to make sure the colon is in

// the correct spot

if (militaryTime.charAt(2)!= ':')//CONDITION TO CHECK COLON POSITION)

{

System.out.println(militaryTime +

" is not a " +

"valid miliary time." );

}

// Check to make sure all other characters

// are digits

else if (! Character.isDigit(militaryTime.charAt(0)))

{

System.out.println(militaryTime +

" is not a " +

"valid miliary time." );

}

else if (! Character.isDigit(militaryTime.charAt(1)))

{

System.out.println(militaryTime +

" is not a " +

"valid miliary time." );

}

else if (! Character.isDigit(militaryTime.charAt(3)))

{

System.out.println(militaryTime +

" is not a " +

"valid miliary time." );

}

else if (! Character.isDigit(militaryTime.charAt(4)))

{

System.out.println(militaryTime +

" is not a " +

"valid miliary time." );

}

else

{

// SEPARATE THE STRING INTO THE HOURS

// AND THE MINUTES, CONVERTING THEM TO

// INTEGERS AND STORING INTO THE

// INSTANCE VARIABLES

this.hours = Integer.parseInt(militaryTime.substring(0,2));

this.minutes = Integer.parseInt(militaryTime.substring(3,5));

// Validate hours and minutes are valid values

if(hours > 21)

{

System.out.println(militaryTime +

" is not a " +

"valid miliary time." );

}

else if(minutes > 59)

{

System.out.println(militaryTime +

" is not a " +

"valid miliary time." );

}

// Convert military time to conventional time

// for afternoon times

else if (hours > 12)

{

hours = hours - 12;

afternoon = true;

System.out.println(this.toString());

}

// Account for midnight

else if (hours == 0)

{

hours = 12;

System.out.println(this.toString());

}

// Account for noon

else if (hours == 12)

{

afternoon = true;

System.out.println(this.toString());

}

// Morning times do not need converting

else

{

System.out.println(this.toString());

}

}

}

}

/\*\*

The toString method returns a conventional time.

@return A conventional time with am or pm.

\*/

public String toString()

{

String am\_pm;

String zero = "";

if (afternoon)

am\_pm = "PM";

else

am\_pm = "AM";

if (minutes < 10)

zero = "0";

return hours + ":" + zero + minutes + " " + am\_pm;

}

}

//-----------------------------------------------------------

//TimeDemo.java

import java.util.Scanner;

/\*\*

This program demonstrates the Time class.

\*/

public class TimeDemo

{

public static void main(String[] args)

{

Scanner keyboard = new Scanner(System.in);

char answer = 'Y';

String enteredTime;

String response;

while ( Character.toUpperCase(answer)=='Y')//CHECK ANSWER AFTER CONVERTING TO CAPITAL)

{

System.out.print("Enter a military time " +

"using the ##:## format: ");

enteredTime = keyboard.nextLine();

Time now = new Time (enteredTime);

System.out.println("Do you want to enter " +

"another (Y/N)? ");

response = keyboard.nextLine();

answer = response.charAt(0);

}

}

}

//-----------------------------------------------------Output

Enter a military time using the ##:## format: 00:00

12:00 AM

Do you want to enter another (Y/N)?

y

Enter a military time using the ##:## format: 12:00

12:00 PM

Do you want to enter another (Y/N)?

y

Enter a military time using the ##:## format: 04:05

4:05 AM

Do you want to enter another (Y/N)?

y

Enter a military time using the ##:## format: 10:15

10:15 AM

Do you want to enter another (Y/N)?

y

Enter a military time using the ##:## format: 23:59

11:59 PM

Do you want to enter another (Y/N)?

y

Enter a military time using the ##:## format: 00:35

12:35 AM

Do you want to enter another (Y/N)?

y

Enter a military time using the ##:## format: 7:56

7:56 is not a valid miliary time.

Do you want to enter another (Y/N)?

y

Enter a military time using the ##:## format: 15:78

15:78 is not a valid miliary time.

Do you want to enter another (Y/N)?

y

Enter a military time using the ##:## format: 08:60

08:60 is not a valid miliary time.

Do you want to enter another (Y/N)?

y

Enter a military time using the ##:## format: 24:00

24:00 is not a valid miliary time.

Do you want to enter another (Y/N)?

y

Enter a military time using the ##:## format: 3e:33

3e:33 is not a valid miliary time.

Do you want to enter another (Y/N)?

y

Enter a military time using the ##:## format: 1:111

1:111 is not a valid miliary time.

Do you want to enter another (Y/N)?

n

Process finished with exit code 0

//-------------------------------------------------------------------

//SecretDemo.java

import java.io.\*;

import java.util.Scanner;

public class SecretDemo {

public static void main(String args [])throws IOException{

File file = new File ("secret.txt");

Scanner inputFile = new Scanner(file);

String fileContents;

int count = 0;

char letter;

StringBuilder strb = new StringBuilder();

fileContents = inputFile.nextLine();

String [] tokens = fileContents.split(" ");

for (String s: tokens){

if (count % 5 == 0){

letter = s.charAt(0);

letter = Character.toUpperCase(letter);

strb.append(letter);

}

count++;

}

System.out.println(strb);

inputFile.close();

}

}

//------------------------OUTPUT

JAVA