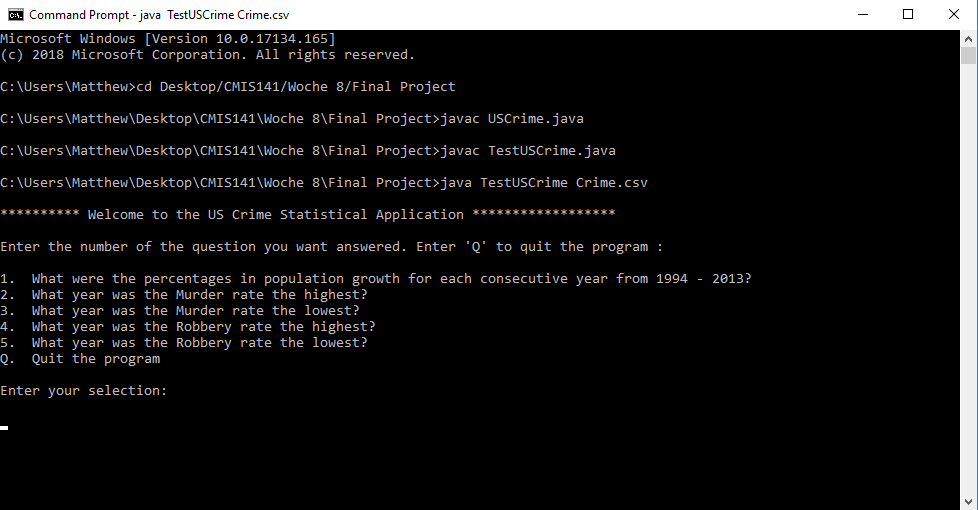
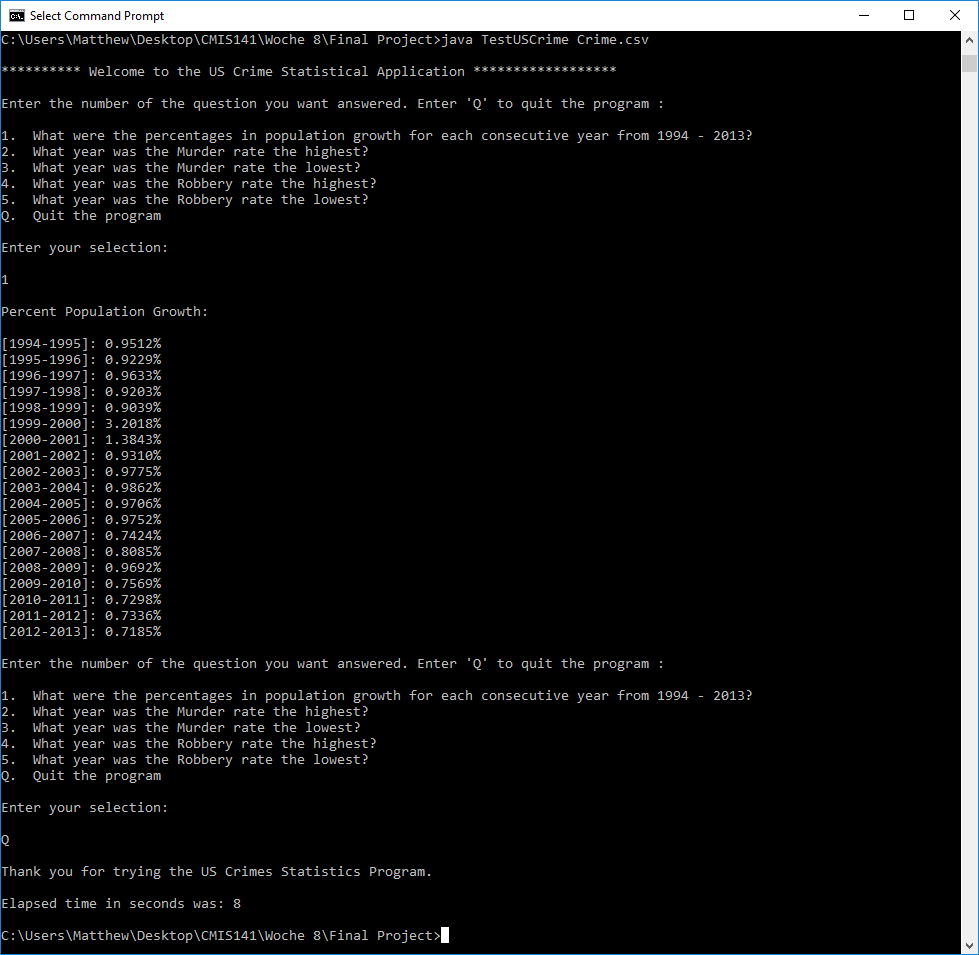
|  |  |  |  |
| --- | --- | --- | --- |
| Input | Expected Output | Actual Output | Pass? |
| Test Case #1  1,  Q | Percent Population Growth:  [1994-1995]: 0.9512%  [1995-1996]: 0.9229%  [1996-1997]: 0.9633%  [1997-1998]: 0.9203%  [1998-1999]: 0.9039%  [1999-2000]: 3.2018%  [2000-2001]: 1.3843%  [2001-2002]: 0.9310%  [2002-2003]: 0.9775%  [2003-2004]: 0.9862%  [2004-2005]: 0.9706%  [2005-2006]: 0.9752%  [2006-2007]: 0.7424%  [2007-2008]: 0.8085%  [2008-2009]: 0.9692%  [2009-2010]: 0.7569%  [2010-2011]: 0.7298%  [2011-2012]: 0.7336%  [2012-2013]: 0.7185%  Thank you for trying the US Crimes Statistics Program.  Elapsed time in seconds was: X | Percent Population Growth:  [1994-1995]: 0.9512%  [1995-1996]: 0.9229%  [1996-1997]: 0.9633%  [1997-1998]: 0.9203%  [1998-1999]: 0.9039%  [1999-2000]: 3.2018%  [2000-2001]: 1.3843%  [2001-2002]: 0.9310%  [2002-2003]: 0.9775%  [2003-2004]: 0.9862%  [2004-2005]: 0.9706%  [2005-2006]: 0.9752%  [2006-2007]: 0.7424%  [2007-2008]: 0.8085%  [2008-2009]: 0.9692%  [2009-2010]: 0.7569%  [2010-2011]: 0.7298%  [2011-2012]: 0.7336%  [2012-2013]: 0.7185%  Thank you for trying the US Crimes Statistics Program.  Elapsed time in seconds was: 8 | Yes |
| Test Case #2  2,  5,  Q | The Murder rate was highest in 1994  The Robbery rate was lowest in 2013  Thank you for trying the US Crimes Statistics Program.  Elapsed time in seconds was: Y | The Murder rate was highest in 1994  The Robbery rate was lowest in 2013  Thank you for trying the US Crimes Statistics Program.  Elapsed time in seconds was: 86 | Yes |
| Test Case #3  3,  6,  4,  Q | The Murder rate was lowest in 2013  Error: Invalid Entry. Please enter a valid menu selection.  The Robbery rate was highest in 1994  Thank you for trying the US Crimes Statistics Program.  Elapsed time in seconds was: Z | The Murder rate was lowest in 2013  Error: Invalid Entry. Please enter a valid menu selection.  The Robbery rate was highest in 1994  Thank you for trying the US Crimes Statistics Program.  Elapsed time in seconds was: 53 | Yes |

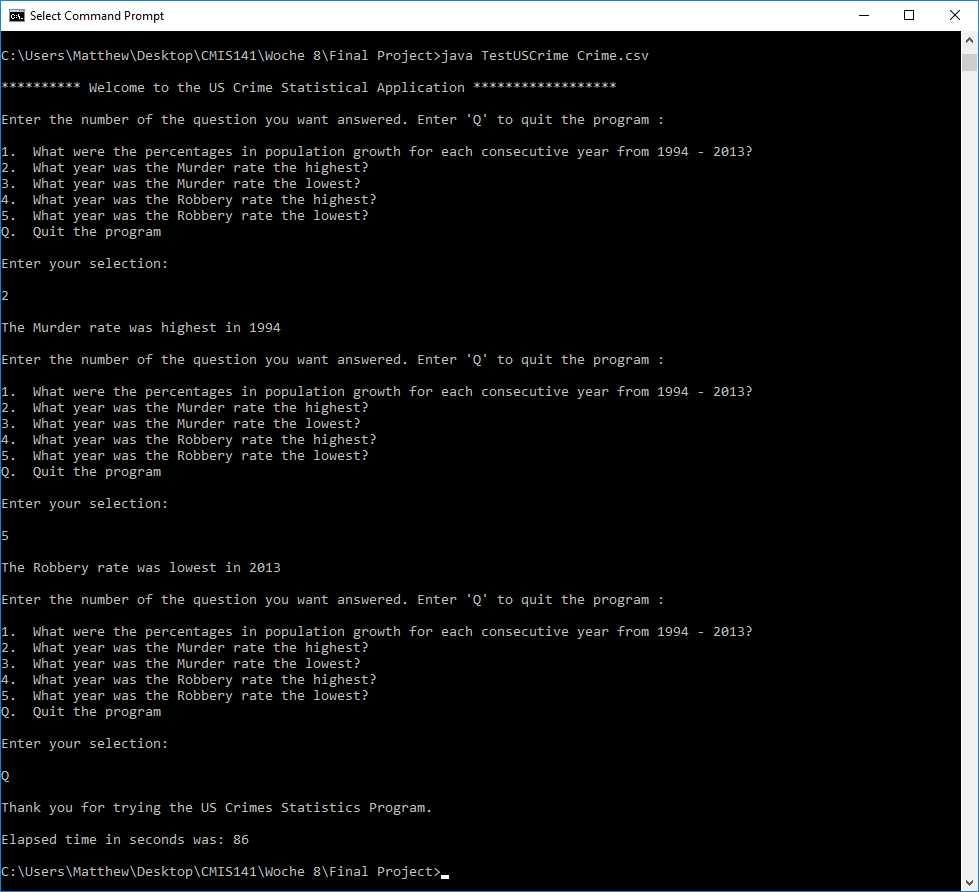
**Screen Capture of me successfully compiling and executing my Java program:**



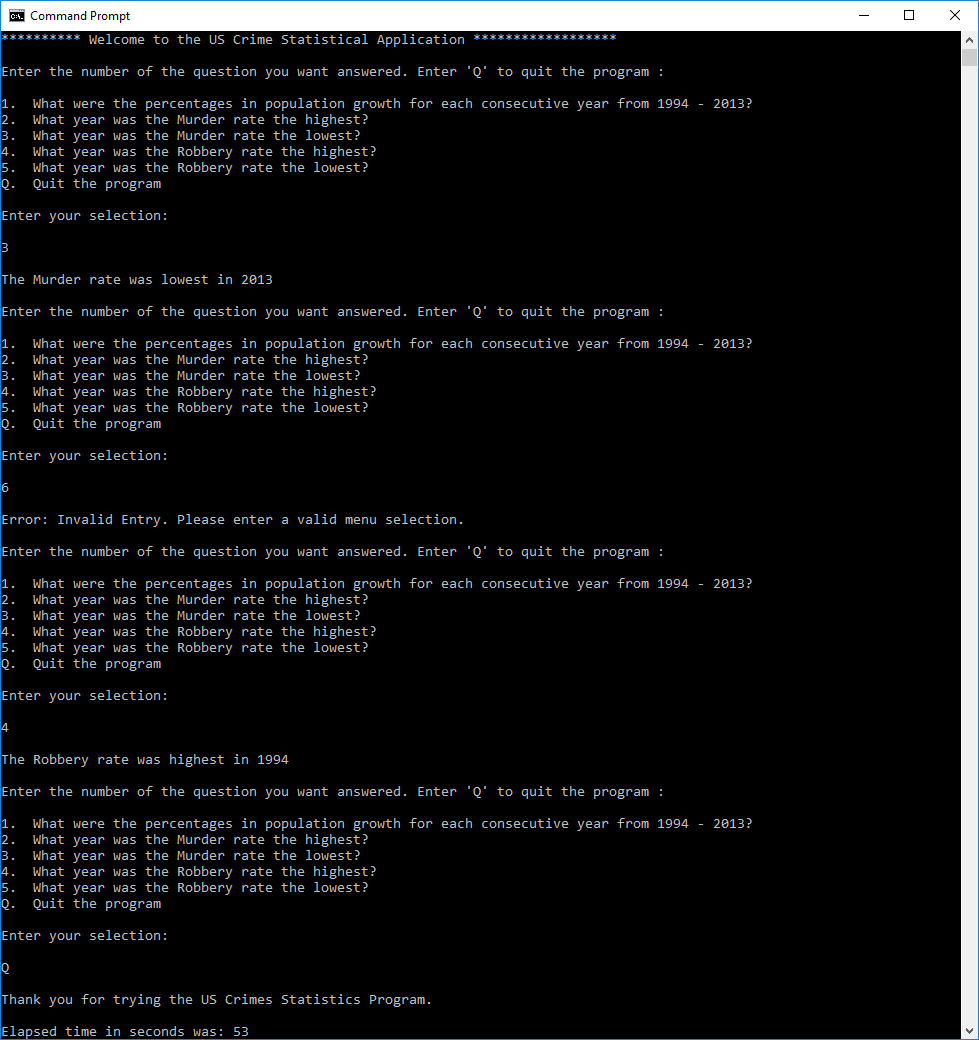
**Screen Capture of Test Case #1:**



**Screen Capture of Test Case #2:**



**Screen Capture of Test Case #3:**



**Source Code:**

**USCrime.java**

/\*

\* Course: CMIS 141

\* File: USCrime.java

\* Author: Andrew H. Rohn

\* Date: 15 July 2018

\* Purpose: This program reads a file containing data related to the U.S. Crime statistics from 1994-2013

\* and displays the population growth in percentages for each consecutive year, the years with

\* the maximum and minimum murder rates, and the maximum and minimum robbery rates.

\*/

public class USCrime {

// Private field definition

private int year;

private int population;

private int murder;

private double murderRate;

private int robbery;

private double robberyRate;

// Split read file lines into fields

public USCrime(String line) {

String[] split = line.split(",");

setYear(Integer.parseInt(split[0]));

setPopulation(Integer.parseInt(split[1]));

setMurder(Integer.parseInt(split[4]));

setMurderRate(Double.parseDouble(split[5]));

setRobbery(Integer.parseInt(split[8]));

setRobberyRate(Double.parseDouble(split[9]));

}

// Fields for methods

public int getYear() {

return year;

}

public void setYear(int year) {

this.year = year;

}

public int getPopulation() {

return population;

}

public void setPopulation(int population) {

this.population = population;

}

public int getMurder() {

return murder;

}

public void setMurder(int murder) {

this.murder = murder;

}

public double getMurderRate() {

return murderRate;

}

public void setMurderRate(double murderRate) {

this.murderRate = murderRate;

}

public int getRobbery() {

return robbery;

}

public void setRobbery(int robbery) {

this.robbery = robbery;

}

public double getRobberyRate() {

return robberyRate;

}

public void setRobberyRate(double robberyRate) {

this.robberyRate = robberyRate;

}

}

**TestUSCrime.java**

/\*

\* Course: CMIS 141

\* File: TestUSCrime.java

\* Author: Andrew H. Rohn

\* Date: 15 July 2018

\* Purpose: This program reads a file containing data related to the U.S. Crime statistics from 1994-2013

\* and displays the population growth in percentages for each consecutive year, the years with

\* the maximum and minimum murder rates, and the maximum and minimum robbery rates.

\*/

import java.io.File;

import java.util.Scanner;

import java.io.FileNotFoundException;

public class TestUSCrime {

// Scanner utility initialization

static Scanner input = new Scanner(System.in);

public static void main(String[] args) {

// Runtime variables

long beginTime = System.currentTimeMillis(); // returns runtime in milliseconds

long endTime;

System.out.println("\n\*\*\*\*\*\*\*\*\*\* Welcome to the US Crime Statistical Application \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

// Prompts user to use command arguments if not done so

if (args.length != 1) {

System.out.println("No file found. Please use command line arguments to use US Crime Data file.\n");

System.out.println("Example: java TestUSCrime Crime.csv");

return;

}

// Calls readFile method for read file in command line argument

USCrime[] data = readFile(args[0]);

// User selection to call methods

String select;

while (true) {

userMenu();

System.out.println("\nEnter your selection: \n");

select = input.next();

switch (select) {

case "1":

System.out.println("\nPercent Population Growth:\n");

popGrowthPercent(data);

System.out.println();

break;

case "2":

System.out.println("\nThe Murder rate was highest in "+getMaxMurderYear(data)+"\n");

break;

case "3":

System.out.println("\nThe Murder rate was lowest in "+getMinMurderYear(data)+"\n");

break;

case "4":

System.out.println("\nThe Robbery rate was highest in "+getMaxRobberyYear(data)+"\n");

break;

case "5":

System.out.println("\nThe Robbery rate was lowest in "+getMinRobberyYear(data)+"\n");

break;

case "Q":

System.out.println("\nThank you for trying the US Crimes Statistics Program.\n");

endTime = System.currentTimeMillis();

System.out.println("Elapsed time in seconds was: " +(endTime - beginTime) / 1000);

return;

default:

System.out.println("\nError: Invalid Entry. Please enter a valid menu selection.\n");

break;

}

}

}

// User menu

static void userMenu() {

System.out.println("Enter the number of the question you want answered. Enter 'Q' to quit the program : \n");

System.out.println("1. What were the percentages in population growth for each consecutive year from 1994 - 2013?");

System.out.println("2. What year was the Murder rate the highest?");

System.out.println("3. What year was the Murder rate the lowest?");

System.out.println("4. What year was the Robbery rate the highest?");

System.out.println("5. What year was the Robbery rate the lowest?");

System.out.println("Q. Quit the program");

}

// Method reads data from read file and saves in USCrime Array

public static USCrime[] readFile(String filename) {

USCrime[] info = new USCrime[20];

int count = 0;

String line;

try {

Scanner inputReader = new Scanner(new File(filename));

inputReader.nextLine();

while (inputReader.hasNext()) {

line = inputReader.nextLine();

info[count] = new USCrime(line);

count++;

}

inputReader.close();

} catch (FileNotFoundException e) {

System.out.println(e);

}

return info;

}

// Method that returns population growth in percentages for each consecutive year

static void popGrowthPercent(USCrime[] data) {

double popGrowth;

for (int i = 0; i < data.length - 1; i++) {

// Equation for population growth

popGrowth = 100 \* (double) (data[i + 1].getPopulation() - data[i].getPopulation()) / data[i].getPopulation();

// Print population growth in string format

System.out.println("[" + data[i].getYear() + "-" + data[i + 1].getYear() + "]: "

+ String.format("%.4f", popGrowth) + "%");

}

}

// Method that returns year with highest murder rate

public static int getMaxMurderYear(USCrime[] data) {

int year = 0;

double maxRate = 0;

for (USCrime value : data) {

if (value.getMurderRate() > maxRate) {

maxRate = value.getMurderRate();

year = value.getYear();

}

}

return year;

}

// Method that returns year with lowest murder rate

public static int getMinMurderYear(USCrime[] data) {

int year = 0;

double minRate = data[0].getMurderRate();

for (USCrime value : data) {

if (value.getMurderRate() < minRate) {

minRate = value.getMurderRate();

year = value.getYear();

}

}

return year;

}

// Method that returns year with highest robbery rate

public static int getMaxRobberyYear(USCrime[] data) {

int year = 0;

double maxRate = 0;

for (USCrime value : data) {

if (value.getRobberyRate() > maxRate) {

maxRate = value.getRobberyRate();

year = value.getYear();

}

}

return year;

}

// Method that returns year with lowest robbery rate

public static int getMinRobberyYear(USCrime[] data) {

int year = 0;

double minRate = data[0].getRobberyRate();

for (USCrime value : data) {

if (value.getRobberyRate() < minRate) {

minRate = value.getRobberyRate();

year = value.getYear();

}

}

return year;

}

}