Trent Giever

CS &141

Ch 7

4/25/2020

Problem # 4

Code:

///////////////////////////////////////////////////////////////////////

/\*

Programmer: Trent Giever

Assignment Chapter: 7

Purpose: larger than n

Date modified: 4/25/2020

IDE/Tool used: NetBeans 8.2

\*/

package ch.pkg7.pkg1;

public class Ch71

{

public static void main(String[] args)

{

double array[] = {10,20,30,40,50,60,70,80,90,100}; //creation of test array

int n = 50; //test n

N myN = new N(array, n); //test object

myN.greaterThan(); //greater than

}

}

////////////////////////////////////////////////////////////////////////

/\*

Programmer: Trent Giever

Assignment Chapter: 7

Purpose: ArrayOperations Class

Date modified: 4/25/2020

IDE/Tool used: NetBeans 8.2

\*/

package ch.pkg7.pkg1;

import static java.lang.System.out;

public class N

{

private double number; //n value

private double array[]; // array passed in

public N(double a[], double n) //constructor passes in the array and an n

{

array = a;

number = n;

}

public void greaterThan() //method to display only the items that are greater than n

{

for(int i = 0; i < array.length; i++)

{

if(array[i] > number)

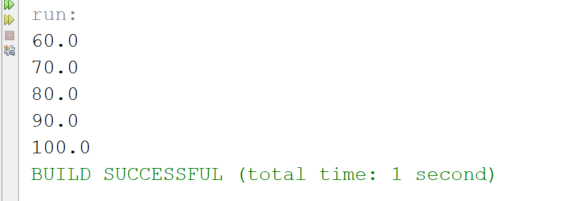
out.println(array[i]);

}

}

}

Snip:



Program # 11

Code:

///////////////////////////////////////////////////////////////////

package ch.pkg7.pkg2;

import java.io.IOException;

import static java.lang.System.out;

public class Ch72

{

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

{

//creation of 4 arrays to test the overloaded methods

int array1[] = {1,2,3,4,5,6,7,8,9,10};

float array2[] = {.1f, .2f, .3f, .4f, .5f, .6f, .7f, .8f, .9f, 1.0f};

double array3[] = {.1,.2,.3,.4,.5,.6,.7,.8,.9,1};

long array4[] = {1L, 2L, 3L, 4L, 5L, 6L, 7L, 8L, 9L, 10L};

ArrayOperations myArray = new ArrayOperations();

//test of total method

out.println("Int total is: " + myArray.getTotal(array1));

out.println("float total is: " + myArray.getTotal(array2));

out.println("double total is: " + myArray.getTotal(array3));

out.println("long total is: " + myArray.getTotal(array4));

//test of the average method

out.println("Int average is: " + myArray.getAverage(array1));

out.println("float average is: " + myArray.getAverage(array2));

out.println("double average is: " + myArray.getAverage(array3));

out.println("long average is: " + myArray.getAverage(array4));

//test of the highest method

out.println("Int highest is: " + myArray.getHighest(array1));

out.println("float highest is: " + myArray.getHighest(array2));

out.println("double highest is: " + myArray.getHighest(array3));

out.println("long highest is: " + myArray.getHighest(array4));

//test of the lowest method

out.println("Int lowest is: " + myArray.getLowest(array1));

out.println("float lowest is: " + myArray.getLowest(array2));

out.println("double lowest is: " + myArray.getLowest(array3));

out.println("long lowest is: " + myArray.getLowest(array4));

}

}

/////////////////////////////////////////////////////////////////////

package ch.pkg7.pkg2;

class ArrayOperations

{

//overloaded getTotal method that returns the total of an array

public static int getTotal(int array[]) //int method

{

int total =0;

for(int i =0; i < array.length; i++)

{

total += array[i];

}

return total;

}

public static float getTotal(float array[]) //float method

{

float total =0f;

for(int i =0; i < array.length; i++)

{

total += array[i];

}

return total;

}

public static double getTotal(double array[]) //double method

{

double total =0;

for(int i =0; i < array.length; i++)

{

total += array[i];

}

return total;

}

public static long getTotal(long array[]) //long method

{

long total =0L;

for(int i =0; i < array.length; i++)

{

total += array[i];

}

return total;

}

//overload getAverage method

public static int getAverage(int array[]) //int method

{

int total =0;

for(int i =0; i < array.length; i++)

{

total += array[i];

}

return total/array.length;

}

public static float getAverage(float array[]) //float method

{

float total =0f;

for(int i =0; i < array.length; i++)

{

total += array[i];

}

return total/array.length;

}

public static double getAverage(double array[]) //double method

{

double total =0;

for(int i =0; i < array.length; i++)

{

total += array[i];

}

return total/array.length;

}

public static long getAverage(long array[]) //long method

{

long total =0L;

for(int i =0; i < array.length; i++)

{

total += array[i];

}

return total/array.length;

}

//overloaded getHighest method

public static int getHighest(int array[])//int method

{

int highest =0;

for(int i =0; i < array.length; i++)

{

if(i ==0)

highest = array[0];

else if(array[i] > highest)

highest = array[i];

}

return highest;

}

public static float getHighest(float array[]) //float method

{

float highest =0f;

for(int i =0; i < array.length; i++)

{

if(i ==0)

highest = array[0];

else if(array[i] > highest)

highest = array[i];

}

return highest;

}

public static double getHighest(double array[]) //double method

{

double highest =0;

for(int i =0; i < array.length; i++)

{

if(i ==0)

highest = array[0];

else if(array[i] > highest)

highest = array[i];

}

return highest;

}

public static long getHighest(long array[]) //long method

{

long highest =0L;

for(int i =0; i < array.length; i++)

{

if(i ==0)

highest = array[0];

else if(array[i] > highest)

highest = array[i];

}

return highest;

}

//overloaded getLowest method

public static int getLowest(int array[])//int method

{

int lowest =0;

for(int i =0; i < array.length; i++)

{

if(i ==0)

lowest = array[0];

else if(array[i] < lowest)

lowest = array[i];

}

return lowest;

}

public static float getLowest(float array[])//float method

{

float lowest =0f;

for(int i =0; i < array.length; i++)

{

if(i ==0)

lowest = array[0];

else if(array[i] < lowest)

lowest = array[i];

}

return lowest;

}

public static double getLowest(double array[])//double method

{

double lowest =0;

for(int i =0; i < array.length; i++)

{

if(i ==0)

lowest = array[0];

else if(array[i] < lowest)

lowest = array[i];

}

return lowest;

}

public static long getLowest(long array[])//long method

{

long lowest =0L;

for(int i =0; i < array.length; i++)

{

if(i ==0)

lowest = array[0];

else if(array[i] < lowest)

lowest = array[i];

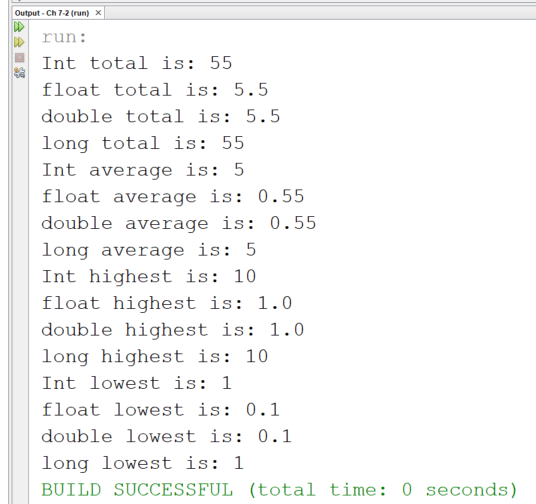
}

return lowest;

}

}

Snip:



Program # 19

Code:

///////////////////////////////////////////////////////////////////

/\*

Programmer: Trent Giever

Assignment Chapter: 7

Purpose: Phone Book

Date modified: 4/25/2020

IDE/Tool used: NetBeans 8.2

\*/

package ch.pkg7.pkg3;

import static java.lang.System.out;

import java.util.ArrayList;

import java.util.Scanner;

public class Ch73

{

public static void main(String[] args)

{

ArrayList<PhoneBook> book = new ArrayList<PhoneBook>(); //creates a array

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

for(int i =1; i <= 10;i++ ) //loop to create 10 contacts

{

out.print("Enter person " + i + " name:");

String name = in.nextLine(); //creates name object

out.print("Enter person " + i + " phone number:");

String num = in.nextLine(); //creates number object

PhoneBook obj = new PhoneBook(name, num); //creates object

book.add(obj);//adds to array

}

for(int i = 0; i < book.size(); i++) //loops through listtes

out.println(book.get(i));

}

}

//////////////////////////////////////////////////////////////////

package ch.pkg7.pkg3;

public class PhoneBook

{

private String name; //holds the persons name

private String phoneNumber; //holds the persons phone number

public PhoneBook() //default constructor

{

name ="";

phoneNumber ="";

}

public PhoneBook(String n, String p) //overload constructor

{

name = n;

phoneNumber = p;

}

public String getName() //returns names

{

return name;

}

public void setName(String n) //sets name

{

name = n;

}

public String getPhoneNumber() // returns phone number

{

return phoneNumber;

}

public void setPhoneNumber(String p)//set phone number

{

phoneNumber = p;

}

@Override

public String toString() //overloaded toString to display the items

{

return "Name: " + name + "\nPhone Number:" + phoneNumber;

}

}

Snip:

