Nathan Gaffney

16-October-2014

CST-183-FA110-14FA-COURSE

This program will create a Health Log Class. It will prompt user for basic health information and then preform calculations to output BMI and other related information.

Step 3.

/\*---------------------  
Created by: NAthan Gaffney  
09-October-2014  
JAVA PROGRAMMING - CST-183-FA110-14FA-COURSE  
This class will create a HealthLog Object  
Errors Handled: Invalid Data  
Dependencies: None  
Methods:  
validData  
HealthLog  
setAge  
setHeight  
setWeight  
setGender  
setRunning  
setSpeed  
setWorking  
setMHR  
setDistance  
getAge  
getHeight  
getWeight  
getGender  
getRunning  
getSpeed  
getWorkingOut  
----------------------\*/  
public class HealthLog  
{  
 private int age;  
 private double height;  
 private double weight;  
 private char gender;  
 private boolean running;  
 private double speed;  
 private double workingOut;  
 private double maxHR;  
   
 /\*\*  
 THis method will preform a check for valid data  
 If valid will return true  
 Else return false  
 \*/  
 public boolean validData()  
 {  
 int t=0;  
 if (age>0 && age < 130)  
 { System.out.println("Age Valid.");}  
 else{System.out.println("Age Invalid.");t++;}  
 if(height > 0 && height < 144)  
 {System.out.println("Height Valid.");}  
 else{System.out.println("Height Invalid.");t++;}   
 if(weight>0 && weight < 1250)  
 {System.out.println("Weight Valid.");}  
 else{System.out.println("Weight Invalid.");t++;}  
 if(gender == 'm' || gender == 'f' || gender == 'M' || gender == 'F')  
 {System.out.println("Gender Valid.");}  
 else{System.out.println("Gender Invalid.");t++;}  
 if(running == true || running == false)  
 {System.out.println("Runner Valid.");}  
 else{System.out.println("Runner Invalid.");t++;}  
 if(speed > 0 && speed < 20)  
 {System.out.println("Speed Valid.");}  
 else{System.out.println("Speed Invalid.");t++;}  
 if(workingOut >= 0 && workingOut < 1440)  
 {System.out.println("Work Out Valid.");}  
 else{System.out.println("Work Out Invalid.");t++;}  
   
 if(t==0){return true;}  
 else{return false;}  
 }  
 /\*\*  
 This is the constructor  
 @ag recieves age  
 @heigh recieves height in inches  
 @weigh recieves weight in pounds  
 @gende recieves gender 'm' or 'f'  
 @runningrecieves boolean if running  
 @spee recieves speed in mph  
 @workingOu recieves time worked out  
 validData checks for valid data  
 \*/  
 public HealthLog(int ag, double heigh, double weigh,char gende,  
 boolean runnin, double spee,double workingOu)  
 {  
 boolean test;  
 age = ag;  
 height = heigh;  
 weight = weigh;  
 gender = gende;  
 running = runnin;  
 speed = spee;  
 workingOut = workingOu;  
 test = validData();  
 if(test == false)  
 {System.out.print("Invalid data. Exiting");  
 System.exit(0);  
 }  
   
   
 }  
   
 /\*\*  
 THis method will set the age.  
 @a revieving age  
 @return return age  
 \*/  
 public void setAge(int a)  
 {  
 age = a;  
 }  
 /\*\*  
 THis method will set the height  
 in inches.  
 @a recieving height  
 @return height  
 \*/  
 public void setHeight(double a)  
 {  
 height = a;  
 }  
 /\*\*  
 THis method will set the weight  
 in pounds  
 @a recieving weight  
 @weight return weight  
 \*/  
 public void setWeight(double a)  
 {  
 weight = a;  
 }  
 /\*\*  
 THis method will set the gender  
 'f' for female  
 'm' for male  
 @a recieving gender char  
 @gender return age  
 \*/  
 public void setGender(char a)  
 {  
 gender = a;  
 }  
 /\*\*  
 This method will determine runner status  
 @runner if runner is true  
 @running stores runner status  
 \*/  
 public void setRunning(boolean runner)  
 {  
 running = runner;  
 }  
 /\*\*  
 THis method will set the spped  
 in mph  
 @a recieving spped  
 @speed return speed  
 \*/  
 public void setSpeed(double a)  
 {  
 speed = a;  
 }  
 /\*\*  
 THis method will set the time working out  
 in minutes  
 @a recieve the time  
 @workingOut output time in hours  
 \*/  
 public void setWorkingOut (double a)  
 {  
 workingOut = a/60;  
 }  
 public int getAge()  
 {  
 return age;  
 }  
 public double getHeight()  
 {  
 return height;  
 }  
 /\*\*  
 COnvert height in inches to CM  
 @return 2.54(conversion factor)  
 \*/  
 public double getCM()  
 {  
 return 2.54 \* height;  
 }  
 public double getWeight()  
 {  
 return weight;  
 }  
 /\*\*  
 COnvert weight in pounds into Kilograms  
 @return 0.453592(conversion factor)  
 \*/  
 public double getKG()  
 {  
 return 0.453592 \* weight;  
 }  
 public char getGender()  
 {  
 return gender;  
 }  
 /\*\*  
 Calculate BMI  
 @weight the weight in pounds  
 @height the height in inches  
 \*/  
 public double getBMI()  
 {  
 return (703\*weight)/(height \*height);  
 }  
 /\*\*  
 This method will determine the BMI class  
 @getBMI will create a BMI index number  
 \*/  
 public String bmiClass()  
 {  
 if (getBMI() < 18.5){return "Underweight";}  
 else if (getBMI() < 24.99){return "Normal";}  
 else if (getBMI() < 29.99){return "Overweight";}  
 else {return "Obese";}  
 }  
 /\*\*  
 This method will calculate  
 the Maximum heart range  
 @age is the age of user  
 \*/  
 public void calcMaxHR()  
 {  
 maxHR = 217-(0.85\*age);  
 }  
 /\*\*  
 This method will calculate  
 the minimum target heart range  
 @maxHR maximum heart range  
 \*/  
 public double minTHR()  
 {  
 return 0.60\*(217-(0.85\*age));  
 }  
 /\*\*  
 This method will calculate the  
 maximum target heart ragne  
 @ maxHR is maximum heart range  
 \*/  
 public double maxTHR()  
 {  
 return 0.80 \*(217-(0.85\*age));  
 }  
 /\*\*  
 THis method wil calculate the distance traveled  
 \*/  
 public double calcDistance()  
 {  
 return speed \* (workingOut/60);  
 }  
 /\*\*  
 THis method will calculate the minimun energy required  
 for healthy living  
 @gender male or female  
 @return formula for the min energy  
 \*/  
 public double minEnergy()  
 {  
 if (gender == 'm')  
 {  
 return 66.5+(13.75\*getKG())+(5.003 \* getCM())-(6.775\*age);  
 }  
 else  
 {  
 return 65.1 + (9.563 \* getKG())+(1.850 \* getCM())-(4.676\* age);  
 }  
 }  
 /\*\*  
 This method will calculate the amount of calories spent  
 @running whether the user is running during activity  
 @distance the distance traveled in miles  
 \*/  
 public double caloriesSpent()  
 {  
 if (running)  
 {  
 return weight \* (0.75) \* (calcDistance());  
 }  
 else  
 {  
 return weight \* (0.53) \* (calcDistance());  
 }  
 }  
  
}

/\*-------  
Prgram Name: HealthDriver Date:09/Oct/2014  
PRogrammer: Nathan Gaffney Class:CST183-Java  
Program Description:  
THis program acts as the driver for the healthlog  
It will prompt the user for basic data  
Then it will preform calculations based on that data  
Errors Handled: None  
Dependencies: HealthLog.class  
Methods: Main  
-------\*/  
import java.util.Scanner;  
public class HealthDriver  
{  
 public static void main(String args[])  
 {  
 Scanner keyboard = new Scanner (System.in);  
 int age;  
 double height;  
 double weight;  
 double speed;  
 double workingOut;  
 char gender;  
 boolean runner;  
 char run;  
   
 System.out.print("Enter your age: ");  
 age = keyboard.nextInt();  
 System.out.print("Enter your height(in inches): ");  
 height = keyboard.nextDouble();  
 System.out.print("Enter your weight(in pounds): ");  
 weight = keyboard.nextDouble();  
 System.out.print("Enter your gender M/F: ");  
 gender = keyboard.next().charAt(0);  
 System.out.print("Do you run? Y/N :");  
 run = keyboard.next().charAt(0);  
 if (run == 'Y' || run == 'y'){runner = true;}  
 else{runner = false;}  
 System.out.println();  
 System.out.print("How fast do you walk/run in mph? :");  
 speed = keyboard.nextDouble();  
 System.out.print("How long do you run/walk in minutes: ");  
 workingOut = keyboard.nextDouble();  
   
 HealthLog hl = new HealthLog(age,height,weight,gender,runner,speed,workingOut);  
 System.out.println("Calories burned during workout: " + hl.caloriesSpent());  
 System.out.println("Maximum Target Heart Range: " + hl.maxTHR());  
 System.out.println("Minimum Target Heart Range: " + hl.minTHR());  
 System.out.println("Body Mass Index: "+hl.getBMI());  
 System.out.println("Body Mass Classification: " + hl.bmiClass());  
 System.out.println("Minimum daily energy requirements: " + hl.minEnergy());  
   
 }  
}

ÏÏ«Ï ----jGRASP exec: java HealthDriver  
ÏÏ§Ï  
¼¼§ÏEnter your age: 45  
¼¼§ÏEnter your height(in inches): 120  
¼¼§ÏEnter your weight(in pounds): 456  
¼¼§ÏEnter your gender M/F: m  
¼¼§ÏDo you run? Y/N :y  
ÏÏ§Ï  
¼¼§ÏHow fast do you walk/run in mph? :12  
¼¼§ÏHow long do you run/walk in minutes: 300  
ÏÏ§ÏAge Valid.  
ÏÏ§ÏHeight Valid.  
ÏÏ§ÏWeight Valid.  
ÏÏ§ÏGender Valid.  
ÏÏ§ÏRunner Valid.  
ÏÏ§ÏSpeed Valid.  
ÏÏ§ÏWork Out Valid.  
ÏÏ§ÏCalories burned during workout: 20520.0  
ÏÏ§ÏMaximum Target Heart Range: 143.0  
ÏÏ§ÏMinimum Target Heart Range: 107.25  
ÏÏ§ÏBody Mass Index: 22.261666666666667  
ÏÏ§ÏBody Mass Classification: Normal  
ÏÏ§ÏMinimum daily energy requirements: 4130.56124