| **Name:** | Mahadev Balla |
| --- | --- |
| **UID:** | 2023300010 |
| **Experiment No.** | 3B |

| **AIM:** | To study method overloading. |
| --- | --- |
| **Program 1** | |
| **PROBLEM STATEMENT :** | Use these functions in a program that inputs a person’s weight,  an estimate for the intensity of physical activity, the number of  minutes spent performing the physical activity, and the number  of calories in one serving of your favorite food.  The program should then calculate and output how many  servings of that food should be eaten per day to maintain the  person’s current weight at the specified activity level. |
| **PROGRAM:** | import java.util.\*;  class cal{  double P, min, intensity, calcon;    cal(){  this.P = P; this.min = min; this.intensity = intensity; this.calcon = calcon;  }  double calccal(double P){  return 70 \* (P / 2.2) \* 0.756;  }  double calccal(double P,double intensity,double min){  return 0.0385 \* intensity \* P \* min;  }    double calccal(double qty, double calcon){  return 0.1 \* qty \* calcon;  }  }  class bw1{  public static void main(String [] arr){  Scanner sc = new Scanner(System.in);    /\*HashMap<String,String> Food = new HashMap<String,String>();  Food.put("Double Cheese Burger","1000 cals/qty");  Food.put("Pizza","2000 cals/qty");  Food.put("Paneer Roll","800 cals/qty");    HashMap<String,String> Phac = new HashMap<String,String>();  Phac.put("Running(10mph)","Intensity : 17");  Phac.put("Running(6mph)","Intensity : 10");  Phac.put("Basketball","Intensity : 8");  Phac.put("Walking(1mph)","Intensity : 1");\*/    double phycal = 0, foodcal = 0, basalmr = 0, calcon = 0, qty = 0, min = 0, totalcalcon = 0, qt1 = 0, qt2 = 0; //hr = 0;  System.out.print("Enter your bodyweight(in pounds) : ");  double P = sc.nextDouble();  if(P<=0){ System.out.println("Invalid input!!"); }  else{  cal a = new cal();  basalmr += a.calccal(P);  System.out.println("Calories required to meet the basal metabolic rate : " + a.calccal(P));    int x = 0;  do{  System.out.print("Calculate the calories required for physical activity\n1. Running(10mph) Intensity : 17\n2. Running(6mph) Intensity : 10\n3. Basketball Intensity : 8\n4. Walking(1mph) Intensity : 1\n5. Exit\nEnter your choice : ");  //System.out.print("Calculate the calories required for physical activity\n" + Phac + "\n5.Exit\nEnter your choice : ");  x = sc.nextInt();  switch(x){  case 1: System.out.print("Enter time(in min) : ");  min = sc.nextDouble();  if(min <= 0){  System.out.println("Invalid input!!");  break; }  //hr = 0.0166667 \* min;  phycal += a.calccal(P,17,min);  System.out.println("Calories required : " + a.calccal(P,17,min));  break;    case 2: System.out.print("Enter time(in min) : ");  min = sc.nextDouble();  if(min <= 0){  System.out.println("Invalid input!!");  break; }  //hr = 0.0166667 \* min;  phycal += a.calccal(P,10,min);  System.out.println("Calories required : " + a.calccal(P,10,min));  break;  case 3: System.out.print("Enter time(in min) : ");  min = sc.nextDouble();  if(min <= 0){  System.out.println("Invalid input!!");  break; }  //hr = 0.0166667 \* min;  phycal += a.calccal(P,8,min);  System.out.println("Calories required : " + a.calccal(P,8,min));  break;    case 4: System.out.print("Enter time(in min) : ");  min = sc.nextDouble();  if(min <= 0){  System.out.println("Invalid input!!");  break; }  //hr = 0.0166667 \* min;  phycal += a.calccal(P,1,min);  System.out.println("Calories required : " + a.calccal(P,1,min));  break;    case 5: System.out.println("Your daily caloric requirement for physical activities : " + phycal);  break;    default : System.out.println("Invalid input!!");  }  }  while(x!=5);    int y = 0;  //do{  System.out.print("Calculate the calories required for digestion\nChoose your favourite food\n1. Double Cheese Burger\n2. Pizza\n3. Paneer Roll\n4. Exit\nEnter your choice : ");  //System.out.print("Calculate the calories required for digestion\nChoose your favourite food\n" + Food + "4.Exit\nEnter your choice : ");  y = sc.nextInt();  switch(y){  case 1: System.out.print("Enter quantity consumed : ");  qty = sc.nextDouble();  if(qty<=0){  System.out.println("Invalid input!!");  break; }  calcon = 1000;  totalcalcon += qty \* calcon;  foodcal += a.calccal(qty,1000);  qt1 = ((foodcal+basalmr+phycal-totalcalcon) \* 1.1)/1000;  qt2 = ((totalcalcon-foodcal+basalmr+phycal) \* 0.9)/1000;  System.out.println("Calories required : " + a.calccal(qty,calcon));  break;    case 2: System.out.print("Enter quantity consumed : ");  qty = sc.nextDouble();  if(qty<=0){  System.out.println("Invalid input!!");  break; }  calcon = 2000;  totalcalcon += qty \* calcon;  foodcal += a.calccal(qty,calcon);  qt1 = ((foodcal+basalmr+phycal-totalcalcon) \* 1.1)/2000;  qt2 = ((totalcalcon-foodcal+basalmr+phycal) \* 0.9)/2000;  System.out.println("Calories required : " + a.calccal(qty,calcon));  break;    case 3: System.out.print("Enter quantity consumed : ");  qty = sc.nextDouble();  if(qty<=0){  System.out.println("Invalid input!!");  break; }  calcon = 800;  totalcalcon += qty \* calcon;  foodcal += a.calccal(qty,calcon);  qt1 = ((foodcal+basalmr+phycal-totalcalcon) \* 1.1)/800;  qt2 = ((totalcalcon-foodcal+basalmr+phycal) \* 0.9)/800;  System.out.println("Calories required : " + a.calccal(qty,calcon));  break;    case 4: System.out.println("Your daily caloric requirement for digestion : " + foodcal);  break;    default : System.out.println("Invalid input!!");  }  //}  //while(y!=4);    System.out.println("Total Daily Caloric Requirement : " + (foodcal+basalmr+phycal));  if(totalcalcon<(foodcal+basalmr+phycal)){  System.out.println("To maintain your body weight, you need to consume " + (foodcal+basalmr+phycal-totalcalcon) + " more calories daily.");  if(y == 1){  System.out.println("Consume " + (int)qt1 + " more burger(s) to maintain your body weight.");  }  else if(y == 2){  System.out.println("Consume " + (int)qt1 + " more pizza(s) to maintain your body weight.");  }  else{  System.out.println("Consume " + (int)qt1 + " more paneer roll(s) to maintain your body weight.");  }  }  else if(totalcalcon>(foodcal+basalmr+phycal)){  System.out.println("To maintain your body weight, you need to reduce your daily calorie consumption by " + (totalcalcon-foodcal+basalmr+phycal) + " calories. ");  if(y == 1){  System.out.println("Consume " + (int)qt2 + " less burger(s) to maintain your body weight.");  }  else if(y == 2){  System.out.println("Consume " + (int)qt2 + " less pizza(s) to maintain your body weight.");  }  else{  System.out.println("Consume " + (int)qt2 + " less paneer roll(s) to maintain your body weight.");  }  }  }  }  } |
| **PROGRAM USING HASHMAP:** | import java.util.\*;  class cal{  double P, min, intensity, calcon;    cal(){  this.P = P; this.min = min; this.intensity = intensity; this.calcon = calcon;  }  double calccal(double P){  return 70 \* (P / 2.2) \* 0.756;  }  double calccal(double P,double intensity,double min){  return 0.0385 \* intensity \* P \* min;  }    double calccal(double qty, double calcon){  return 0.1 \* qty \* calcon;  }  }  class bw{  public static void main(String [] arr){  Scanner sc = new Scanner(System.in);  HashMap<Integer,String> Food = new HashMap<Integer,String>();  Food.put(1,"Double Cheese Burger");  Food.put(2,"Pizza");  Food.put(3,"Paneer Roll");  Food.put(4,"Exit");    HashMap<Integer,String> Phac = new HashMap<Integer,String>();  Phac.put(1,"Running(10mph)");  Phac.put(2,"Running(6mph)");  Phac.put(3,"Basketball");  Phac.put(4,"Walking(1mph)");  Phac.put(5,"Exit");    double phycal = 0, foodcal = 0, basalmr = 0, calcon = 0, qty = 0, min = 0, totalcalcon = 0, qt1 = 0, qt2 = 0; //hr = 0;  System.out.print("Enter your bodyweight(in pounds) : ");  double P = sc.nextDouble();  if(P<=0){ System.out.println("Invalid input!!"); }  else{  cal a = new cal();  basalmr += a.calccal(P);  System.out.println("Calories required to meet the basal metabolic rate : " + a.calccal(P));    int x = 0;  do{  //System.out.print("Calculate the calories required for physical activity\n1. Running(10mph) Intensity : 17\n2. Running(6mph) Intensity : 10\n3. Basketball Intensity : 8\n4. Walking(1mph) Intensity : 1\n5. Exit\nEnter your choice : ");  System.out.print("Calculate the calories required for physical activity\n" + Phac + "\nEnter your choice : ");  x = sc.nextInt();  switch(x){  case 1: System.out.print("Enter time(in min) : ");  min = sc.nextDouble();  if(min <= 0){  System.out.println("Invalid input!!");  break; }  //hr = 0.0166667 \* min;  phycal += a.calccal(P,17,min);  System.out.println("Calories required : " + a.calccal(P,17,min));  break;    case 2: System.out.print("Enter time(in min) : ");  min = sc.nextDouble();  if(min <= 0){  System.out.println("Invalid input!!");  break; }  //hr = 0.0166667 \* min;  phycal += a.calccal(P,10,min);  System.out.println("Calories required : " + a.calccal(P,10,min));  break;  case 3: System.out.print("Enter time(in min) : ");  min = sc.nextDouble();  if(min <= 0){  System.out.println("Invalid input!!");  break; }  //hr = 0.0166667 \* min;  phycal += a.calccal(P,8,min);  System.out.println("Calories required : " + a.calccal(P,8,min));  break;    case 4: System.out.print("Enter time(in min) : ");  min = sc.nextDouble();  if(min <= 0){  System.out.println("Invalid input!!");  break; }  //hr = 0.0166667 \* min;  phycal += a.calccal(P,1,min);  System.out.println("Calories required : " + a.calccal(P,1,min));  break;    case 5: System.out.println("Your daily caloric requirement for physical activities : " + phycal);  break;    default : System.out.println("Invalid input!!");  }  }  while(x!=5);    int y = 0;  //do{  //System.out.print("Calculate the calories required for digestion\nChoose your favourite food\n1. Double Cheese Burger\n2. Pizza\n3. Paneer Roll\n4. Exit\nEnter your choice : ");  System.out.print("Calculate the calories required for digestion\nChoose your favourite food\n" + Food + "\nEnter your choice : ");  y = sc.nextInt();  switch(y){  case 1: System.out.print("Enter quantity consumed : ");  qty = sc.nextDouble();  if(qty<=0){  System.out.println("Invalid input!!");  break; }  calcon = 1000;  totalcalcon += qty \* calcon;  foodcal += a.calccal(qty,1000);  qt1 = ((foodcal+basalmr+phycal-totalcalcon) \* 0.9)/1000;  qt2 = ((totalcalcon-foodcal+basalmr+phycal) \* 0.9)/1000;  System.out.println("Calories required : " + a.calccal(qty,calcon));  break;    case 2: System.out.print("Enter quantity consumed : ");  qty = sc.nextDouble();  if(qty<=0){  System.out.println("Invalid input!!");  break; }  calcon = 2000;  totalcalcon += qty \* calcon;  foodcal += a.calccal(qty,calcon);  qt1 = ((foodcal+basalmr+phycal-totalcalcon) \* 0.9)/2000;  qt2 = ((totalcalcon-foodcal+basalmr+phycal) \* 0.9)/2000;  System.out.println("Calories required : " + a.calccal(qty,calcon));  break;    case 3: System.out.print("Enter quantity consumed : ");  qty = sc.nextDouble();  if(qty<=0){  System.out.println("Invalid input!!");  break; }  calcon = 800;  totalcalcon += qty \* calcon;  foodcal += a.calccal(qty,calcon);  qt1 = ((foodcal+basalmr+phycal-totalcalcon) \* 0.9)/800;  qt2 = ((totalcalcon-foodcal+basalmr+phycal) \* 0.9)/800;  System.out.println("Calories required : " + a.calccal(qty,calcon));  break;    case 4: System.out.println("Your daily caloric requirement for digestion : " + foodcal);  break;    default : System.out.println("Invalid input!!");  }  //}  //while(y!=4);    System.out.println("Total Daily Caloric Requirement : " + (foodcal+basalmr+phycal));  if(totalcalcon<(foodcal+basalmr+phycal)){  System.out.println("To maintain your body weight, you need to consume " + (foodcal+basalmr+phycal-totalcalcon) + " more calories daily.");  if(y == 1){  System.out.println("Consume " + (int)qt1 + " more burger(s) to maintain your body weight.");  }  else if(y == 2){  System.out.println("Consume " + (int)qt1 + " more pizza(s) to maintain your body weight.");  }  else{  System.out.println("Consume " + (int)qt1 + " more paneer roll(s) to maintain your body weight.");  }  }  else if(totalcalcon>(foodcal+basalmr+phycal)){  System.out.println("To maintain your body weight, you need to reduce your daily calorie consumption by " + (totalcalcon-foodcal+basalmr+phycal) + " calories. ");  if(y == 1){  System.out.println("Consume " + (int)qt2 + " less burger(s) to maintain your body weight.");  }  else if(y == 2){  System.out.println("Consume " + (int)qt2 + " less pizza(s) to maintain your body weight.");  }  else{  System.out.println("Consume " + (int)qt2 + " less paneer roll(s) to maintain your body weight.");  }  }  }  }  } |
| **RESULT:** | |
| **Program 2** | |
| **PROBLEM STATEMENT :** | Create a Test class with a data double base, int power, int  logBase, int argument.  Create a default, no-argument constructor which sets the default  value of all variables to 2.  There are 2 overloaded functions:  1. double calculate (double base, int power)  This function returns the value when \*base\* is raised to  \*power\*  For example: calculate (3.0, 2) returns the value of 3.0 raised to  2 i.e., 9.0  2. double calculate (int logBase, int argument)  This function returns the value of the log of \*argument\* to the  base \*logBase\*.  For example: calculate (3, 9) returns log of 9 to the base 3 i.e.,  2.0  Create a main method in a separate class to call the above  functions with the following inputs:  1. calculate (2, 4) |
| **PROGRAM:** | import java.util.\*;  class Test{  double base;  int power, logbase, argument;    Test(){  this.base = 2; this.power = 2; this.logbase = 2; this.argument = 2;  }    double calculate(double base, int power){  return Math.pow(base,power);  }  double calculate(int logbase, int argument){  return (Math.log(argument)/Math.log(logbase));  }  }  class calc{  public static void main(String [] arr){  Scanner sc = new Scanner(System.in);  int x = 0;  do{  System.out.print("Choose an operation to pereform -\n1. Exponential function\n2. Logarithmic function\n3. Exit\nEnter your choice : ");  x = sc.nextInt();  switch(x){  case 1: System.out.print("Enter base : ");  double base = sc.nextDouble();  System.out.print("Enter power : ");  int power = sc.nextInt();  /\*if(base == 0 && power == 0){  System.out.println("Value not defined.");  }\*/  Test a = new Test();  System.out.println("The value of " + base +" to the power " + power + " is : " + a.calculate(base,power));  break;    case 2: System.out.print("Enter logbase : ");  int logbase = sc.nextInt();  System.out.print("Enter argument : ");  int argument = sc.nextInt();  if(logbase == 0 && argument == 0){  System.out.print("Value not defined.");  }  if(argument<0){  System.out.print("Value not defined.");  }  Test b = new Test();  System.out.println("The value of log of " + argument + " to the base " + logbase + " is : " + b.calculate(logbase,argument));  break;  case 3: break;  default: System.out.println("Invalid input!!");  }  }  while(x!=3);  }  } |
| **RESULT:** | |
| **CONCLUSION:** | Studied the implementation of overloaded methods. |