1.  
**Code :**

package com.iem.lab4;

import static java.lang.System.exit;

import java.util.Scanner;

class triangle

{

int base;

int height;

triangle()

{

base=0;

height=0;

}

triangle(int x,int y)

{

base=x;

height=y;

}

double getArea()

{

return 0.5\*base\*height;

}

void show()

{

System.out.println("\nbase:"+base+"\nheight:"+height+"\narea:"+0.5\*base\*height);

}

public static void compare(triangle first, triangle second)

{

if(first.getArea()>second.getArea())

System.out.println("T1 is LARGER");

else if(first.getArea()<second.getArea())

System.out.println("T2 is LARGER");

else

System.out.println("T1 & T2 are EQUAL");

}

void check(){

if (base<=0 || height<=0){

System.out.println("ERROR. Base or Height cannot be negetive / 0");

exit(0);}

}

public static void main(String[] ar)

{

Scanner sc = new Scanner(System.in);

System.out.println("Enter dimensions for T1 (BASE, HEIGHT) :");

int x=sc.nextInt();

int y=sc.nextInt();

triangle t1=new triangle(x,y);

t1.check();

System.out.println("Enter dimensions for T2 (BASE, HEIGHT) :");

x=sc.nextInt();

y=sc.nextInt();

triangle t2=new triangle(x,y);

t2.check();

t1.show();

t2.show();

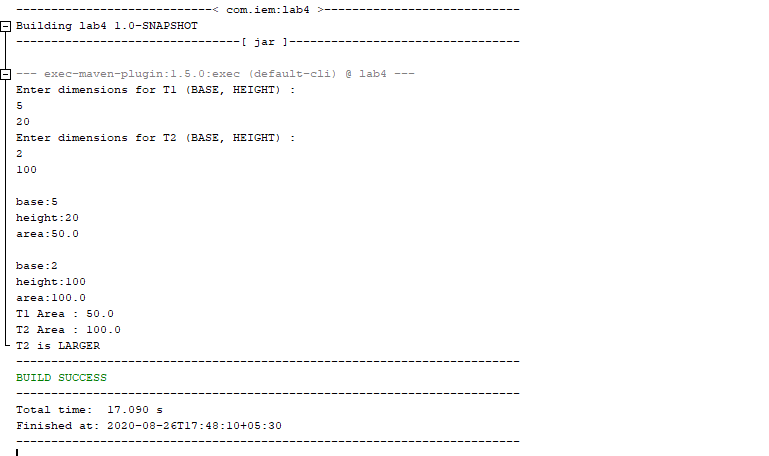
System.out.println("T1 Area : "+t1.getArea());

System.out.println("T2 Area : "+t2.getArea());

triangle.compare(t1,t2);

}

}

**OUTPUT :**  
2.

**Code :**

import java.util.Scanner;

public class IFCSManager {

private Equipment[] eqpList;

private int length;

public static void main(String[] args) {

boolean quit = false;

Scanner sc = new Scanner(System.in);

IFCSManager myMgr = new IFCSManager();

// while loop for user input

while(!quit){

System.out.println("Enter choice number:");

System.out.println("1. Insert\n2. Remove\n3. Report");

System.out.print("4. Display\n5. Quit:");

int choice = sc.nextInt();

switch(choice){

case 1:{

System.out.print("Enter id:");

String id = sc.next();

System.out.print("Enter description:");

String desc = sc.next();

if(myMgr.insert(new Equipment(id, desc)))

System.out.println("Equipment added to Lab");

else

System.out.println("Equipment cannot be added");

}break;

case 2:{

System.out.print("Enter id:");

String id = sc.next();

if(myMgr.remove(id))

System.out.println("Equipment removed from Lab");

else

System.out.println("Equipment with "+id+

" cannot be found.");

}break;

case 3:{

System.out.print("Enter id:");

String id = sc.next();

if(myMgr.find(id))

System.out.println("Equipment available in Lab");

else

System.out.println("Equipment not in Lab");

}break;

case 4:{

myMgr.display();

}break;

case 5:{

System.out.println("Exiting...");

quit = true;

}break;

default:

System.out.println("Invalid Choice");

}

}

}

/\*\*

\* Default constructor, initialises Equipment list

\*/

public IFCSManager() {

eqpList = new Equipment[10];

length = 0;

}

/\*\*

\* Inserts the Equipment into the list

\* @param eqp the Equipment instance to be inserted into the list

\* @return true if Equipment is successfully inserted

\*/

public boolean insert(Equipment eqp) {

if(eqp.id.equals("null")){

System.out.println("Error - id cannot be null");

return false;

}

if(eqp.description.equals("null")){

System.out.println("Error - description cannot be null");

return false;

}

if(length + 1 < 11){

eqpList[length] = eqp;

length++;

return true;

}

return false;

}

/\*\*

\* Removes the Equipment instance, specified by its id, from the list

\* @param id

\* @return true if Equipment is successfully removed

\*/

public boolean remove(String id) {

if(id.equals("null")){

System.out.println("Error - id cannot be null");

return false;

}

for(int i=0; i<length; i++){

if(eqpList[i].id.equals(id)){

for(int j=i+1; j<length; j++){

eqpList[j-1] = eqpList[i];

}

length--;

return true;

}

}

return false;

}

/\*\*

\* Locates the Equipment instance with the specified id

\* @param id

\* @return

\*/

public boolean find(String id) {

if(id.equals("null")){

System.out.println("Error - id cannot be null.");

return false;

}

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

if(eqpList[i].id.equals(id))

return true;

return false;

}

public void display() {

for(int i=0; i<length; i++){

System.out.println("id="+eqpList[i].id+", "+

"desc="+eqpList[i].description);

}

}

}

class Equipment{

String id, description;

public Equipment(String id, String desc){

this.id = id;

this.description = desc;

}

String getId(){

return this.id;

}

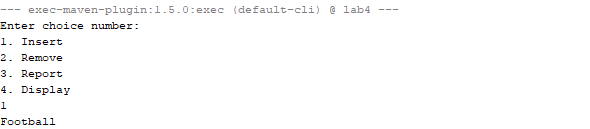
String getDesc(){

return this.description;

}

}

**OUTPUT :**

****