CSE111:

**Design the Club class:**

| **Driver** | **Output** |
| --- | --- |
| public class ClubTester {  public static void main(String[] args) {  Club club1 = new Club();  System.*out*.println("1=================");  System.*out*.println(club1.approveClub("Makers Club",4,10000, 3));  System.*out*.println("2=================");  System.*out*.println(club1.approveClub("Makers Club",10,10000, 3));  System.*out*.println("3=================");  club1.info();  System.*out*.println("4=================");  club1.createEvent("Exhibit", 1099, 5);  System.*out*.println("5=================");  club1.createEvent("Impromptu", 8700, 6);  System.*out*.println("6=================");  club1.recruitMember(5);  System.*out*.println("7=================");  club1.createEvent("Impromptu", 8700, 6);  System.*out*.println("8=================");  club1.info();  System.*out*.println("9=================");  club1.createEvent("Potluck", 1200, 3);  System.*out*.println("10=================");  club1.createEvent("Potluck", 100, 3);  System.*out*.println("11=================");  club1.info();  System.*out*.println("12=================");  club1.createEvent("Speech", 100, 2);  System.*out*.println("13=================");  club1.endEvent("Exhibit");  System.*out*.println("14=================");  club1.info();  System.*out*.println("15=================");  club1.createEvent("Speech",100, 2);  }  } | 1=================  A club must have at least 5 members  2=================  New club, Makers Club, created with 10 members.  3=================  Name of club: Makers Club  Non-working members: 10  Current Budget: 10000.0  No events yet.  4=================  New event, "Exhibit" has started!  5 members are now working.  5=================  Need 1 more member(s) to arrange.  6=================  New members recruited  Total non-working members now are 10.  7=================  New event, "Impromptu" has started!  6 available members are now working.  8=================  Name of club: Makers Club  Non-working members: 4  Current Budget: 201.0  Events:  Exhibit  Impromptu  9=================  Not enough budget.  10=================  New event, "Potluck" has started!  3 members are now working.  11=================  Name of club: Makers Club  Non-working members: 1  Current Budget: 101.0  Events:  Exhibit  Impromptu  Potluck  12=================  Need 1 more member(s) to arrange.  13=================  Exhibit has ended!  5 members are free now.  14=================  Name of club: Makers Club  Non-working members: 6  Current Budget: 101.0  Events:  Impromptu  Potluck  15=================  New event, "Speech" has started!  2 members are now working. |

**public class Club {**

**public String name;**

**public int member;**

**public double budget;**

**public int events;**

**public String eventName[];**

**public int eventMem[];**

**public String approveClub(String n, int m, double budget, int cap) {**

**if (m >= 5) {**

**name = n;**

**member = m;**

**this.budget = budget;**

**events = cap;**

**eventName = new String[cap];**

**eventMem = new int[cap];**

**return "New club, "+name+", created with "+member+" members.";**

**}**

**else {**

**return "A club must have at least 5 members";**

**}**

**}**

**public void info(){**

**System.*out*.println("Name of the club: "+name);**

**System.*out*.println("Non-working Members: "+member);**

**System.*out*.println("Current Budget: "+budget);**

**if(events==eventName.length)**

**System.*out*.println("No events yet.");**

**else{**

**System.*out*.println("Current events: ");**

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

**if(eventName[i]!=null){**

**System.*out*.println(eventName[i]);**

**}**

**}**

**}**

**}**

**public void createEvent(String ename, double ebud, int emem){**

**if(budget < ebud){**

**System.*out*.println("Not enough budget.");**

**}**

**else if(member<emem){**

**System.*out*.println("Need "+(emem-member)+" more member(s) to arrange.");**

**}**

**else if (events==0) {**

**System.*out*.println("Only " + eventName.length + " events can be created.");**

**}**

**else{**

**System.*out*.println("New event, \""+ename+ "\"has started!\n" + emem +" out of "+member+" available members are now working.");**

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

**if(eventName[i]==null){**

**eventName[i]=ename;**

**eventMem[i] = emem;**

**member = member-emem;**

**budget = budget-ebud;**

**events--;**

**break;**

**}**

**}**

**}**

**}**

**public void recruitMember(int m){**

**member += m;**

**System.*out*.println("New members recruited\n" +**

**"Total non-working members now are "+member);**

**}**

**public void endEvent(String ename){**

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

**if(eventName[i]!=null && eventName[i].equals(ename)){**

**eventName[i] = null;**

**member = member+eventMem[i];**

**eventMem[i] = 0;**

**events++;**

**break;**

**}**

**}**

**System.*out*.println(ename+" has ended!\n" +**

**member+" members are free now.");**

**}**

**}**

| **1** | **public class tracing1 {** |
| --- | --- |
| **2** | **public static void main(String[] args) {** |
| **3** | **Test m = new Test();** |
| **4** | **m.n = m.m = 5;** |
| **5** | **Test n = new Test();** |
| **6** | **n.m = m.metA(2);** |
| **7** | **n.n = n.metA(4);** |
| **8** | **System.out.println(m.n+m.m+" "+n.m+" "+n.n);** |
| **9** | **}** |
| **10** | **}** |
| **11** |  |
| **12** | **class Test {** |
| **13** | **int m, n = 1;** |
| **14** |  |
| **15** | **int metA(int n){** |
| **16** | **n += m + 3;** |
| **17** | **int s = n+ this.n;** |
| **18** | **if (s%2 == 0) return s;** |
| **19** | **Test t = new Test();** |
| **20** | **t.n = (++this.m) - (++m) + t.m;** |
| **21** | **this.n = n + t.metA(t.m);** |
| **22** | **System.out.printf("%d %d %d\n", m, n, s);** |
| **23** | **return s+this.n;** |
| **24** | **}** |
| **25** | **}** |