**Question:**

Mappa animation studio produces animated movies and aims to properly manage their projects and staff. The studio works on many different movies at once but ensures that the staff is properly assigned to each of their projects. Usually, each one of their movie projects are assigned a staff member as project lead, and another staff member as chief animator.

Every movie the studio makes has a total budget that is allocated at the time when that movie project is initially started. This budget is always fixed and Mappa does not allow it to be changed at all for a movie. During movie creation process, the job of project lead is to make sure that the current cost of the movie never exceeds its given allocated budget. The project lead immediately alerts everyone if the budget is exceeded. The chief animator is normally concerned with animation and their job is to test the animation quality. But testing quality is costly and each time the chief animator tests the quality of animation, the current cost increases by PKR 50,000.

Apart from project lead and chief animator, all of the movie projects that Mappa start have a movie title and movie id is given to them, also total budget and current cost is maintained for the movie.

Moreover, the staff members, including project lead and chief animator, at Mappa obviously charge different salaries. For their financial record, Mappa keep track of the total amount they are paying in salaries to all of their project leads and chief animators combined.

Implement the above scenario in code using the object-oriented concepts.

**Solution:**

**// Part A and B**class Movie  
{

Staff ProjectLead;  
 Staff ChiefAnimator;  
 const double budget;  
 double CurrentCost;  
 static double totalBudget;

string MovieID;  
 string MovieTitle;

public:  
 Movie (Staff pl, Staff ca, double ct, double budget, string id, string title) : budget(budget)  
 {

CurrentCost = ct;  
ProjectLead = pl;  
ChiefAnimator = ca;  
MovieID = id;  
MovieTitle = title;  
totalBudget += budget;

}

void Production ()  
{

cout << “Task is in production”;  
ProjectLead.TrackProject (CurrentCost);  
ChiefAnimator.Animate (CurrentCost);

} };  
double Movie::totalBudget = 0;

class Staff  
{

double salary;  
string type;  
static double totalSalary;

public:  
 Staff (double salary, string type)  
 {

this->salary = salary;  
 this->type = type;  
 totalSalary += salary;

}

double TrackProject (double currCost)  
 {  
 if (currCost > 200000)  
 cout << “Warning: The cost is exceeding budget.”; return currCost;

}

double Animate (double currCost)  
 {  
 // animates movie   
 currCost += 10000; return currCost;

} };  
double Staff::totalSalary = 0;

**// Part c**

**Staff**

- salary: double  
- type: string  
- totalSalary: double

+Staff(double, string)  
+Production() : void  
+TrackProject(double): double  
+Animate(double): double

**Movie**

- ProjectLead: Staff  
- ChiefAnimator: Staff  
- BUDGET: double  
- CurrentCost: double  
- totalBudget: double  
- MovieID: string  
- MovieTitle: string

+Movie(Staff,Staff,double,double,string,string)  
+Production() : void