

Interface Finance

```
public interface Finance {  
    public abstract double calcBudget();  
    public abstract void display();  
}
```

Class Ministry

```
public abstract class Ministry implements Finance{  
  
    private String name;  
    protected int nbEmployees;  
  
    public Ministry(String name, int nbEmployees) {  
        this.name = name;  
        this.nbEmployees = nbEmployees;  
    }  
    public String getName() {  
        return name;  
    }  
    public int getNbEmployees() {  
        return nbEmployees;  
    }  
  
    public abstract double calcBudget();  
  
    public void display(){  
        System.out.println("Name: " + name  
            + "\nNbEmployees: " + nbEmployees);  
    }  
}
```

Class Executive

```
public class Executive extends Ministry{
    private double expenses;
    public Executive(String name, int nbEmployees, double expenses) {
        super(name, nbEmployees);
        this.expenses = expenses;
    }
    public double getExpenses() { return expenses; }
    public double calcBudget() {
        return expenses + nbEmployees * 1.5;
    }
    @Override
    public void display(){
        super.display();
        System.out.println("Expenses: " + expenses);
    }
}
```

Class Administrative

```
public class Administrative extends Ministry{
    private double balance;
    public Administrative(String name, int nbEmployees, double balance) {
        super(name, nbEmployees);
        this.balance = balance;
    }
    public double getBalance() {return balance;}
    public double calcBudget() {
        return nbEmployees * 10000 - balance;
    }
    @Override
    public void display(){
        super.display();
        System.out.println("Balance: " + balance);
    }
}
```

Class Other

```
public class Other extends Ministry{
    public Other(String name, int nbEmployees){
        super(name, nbEmployees);
    }
    public double calcBudget() {
        return nbEmployees * 10000;
    }
}
```