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


public abstract class Ministry implements Finance{
protected int nbEmployee;
private String name;
Ministry(String name, int nbEmployee)
{
this.name = name;
this.nbEmployee = nbEmployee;
}
Ministry(Ministry m)
{
name = m.name;
nbEmployee = m.nbEmployee;
}

public int getNbEmployee() {
return nbEmployee;
}
public String getName() {
return name;
}
public void display() {
System.out.print(name + nbEmployee);
}
}

public class Executive extends Ministry{
private double expenses;
Executive(String name, int nbEmployee, double expenses)
{
super(name,nbEmployee);
this.expenses = expenses;
}
```

```
Executive(Executive e) {
super(e);
expenses = e.expenses;
}

public double calculateBudget() {
return expenses + nbEmployee * 1.5;
}
public double getExpenses() {
return expenses;
}
}
```

```
public class Government {
private String name;
private Ministry arMin[];
private int nb;
Government(String name, int size)
{
this.name = name;
arMin = new Ministry[size];
nb = 0;
}
public void addMinistry(Ministry m)
{
if(nb >= arMin.length)
return;

if(m instanceof Executive)
      arMin[nb] = new Executive( (Executive)m );
else if(m instanceof Administrative)
      arMin[nb] = new Administrative( (Administrative)m );
else
      arMin[nb] = new Other( (Other)m );
nb++;
}
```

```
public double averageOfBudget()
{
if(nb == 0)
return 0;
double sum =0;
for(int i=0; i<nb; i++)
{
      sum+=arMin[i].calculateBudget();
}
return sum/nb;
}


public int countExecutive(double e)
{
int count = 0;
for(int i=0;i<nb;i++)
if(arMin[i] instanceof Executive)
{ //alt: if(((Executive) arMin[i]).getExpenses > e)
Executive x = (Executive) arMin[i];
if(x.getExpenses() > e)
count++;
}
return count;
}

public int getExecutives(double e, Executive ae[])
{
double avg = averageOfBudget();
int j=0;
for(int i=0; i<nb; i++)
{
if(arMin[i] instanceof Executive)
if(arMin[i].calculateBudget() > avg)
{
Executive x = (Executive) arMin[i];
if(x.getExpenses() > e)
{
ae[j]=x;
j++;
}
}
}
return j;



}
}
```