Solution File

Question 1

a)

class Business {
protected:
Office * offices;
};
	class Brokerage: public
	Business {
	};
		class Marketing: public
		Business {
		};

class Leasing: public
Business {
};
	class Office {
	Employee * emps;
	float revenue;
	string code, address, phone;
	};
		class Employee {
		string empID;
		float salary;
		int hours; };

b)

Business () {
offices = new Office;
}
Business (int n) {
string c, a, p;
offices = new Office[n];
for(int i = 0; i < n; i++)
cin >> c >> a >> p;
offices[i] = Office(c,a,p);
}
	Brokerage (int n_offices):
	Business(n_offices) {
	}
		Marketing (int n_offices):
		Business(n_offices) {
		}

Leasing (int n_offices):
Business(n_offices) {
}
	Office (string c, string a,
	string p, int n) {
	revenue = 0;
	code = c;
	address = a;
	phone = p;
	string e; float s; int h;
	emps = new Employee[n];
	for(int i = 0; i < n; i++)
	cin >> e >> s >> h;
	emps[i] = Employee(e,s,h);
	} }
		Employee (string e, float
		s, int h) {
		empID = e;
		salary = s;
		hours = h;
		}

c)

~office () {
delete emps[];
}
	~Brokerage () {
	delete offices[];
	}

d)

```
// paid_salaries must be a static member variable of Office
ViewAmount(Office * offices, int n)
{
float total = 0;
for(int i = 0; i < n; i++)
  total+= offices[i].paid_salaries;
}
cout << total;
}
```

e)

```
TaxReport(Business * b)
{
if(typeid(b).name() == “Brokerage”)
  cout << b.getrevenue()/2;
else if (typeid(b).name() == “Marketing”)
  cout <<
(b.offices[0].getrevenue()+b.offices[1].getrevenue()+;b.offices[2].getrevenue())/5;

else if(typeid(b).name() == “Leasing”) {
float total = 0;
for(int i = 0; i < n; i++)
  total+= b.offices[i].revenue;
}
  cout << total() – ViewAmount();
}}
```