**BACS 387 – Quiz 1 [Fall 2020] [11/13/20]**

This quiz is worth 30 points. You need to create a list of classes that best represent the data below, you must demonstrate each of abstraction (3 points), encapsulation (10 points) and inheritance (10 points).

A bank member has a global balance (this balance is the sum of all balances in his membership). A membership must have 3 types of accounts, as follows:

* Checking account:
  + Member must define % of balance allocation
* Savings account:
  + Member must define % of balance allocation
  + Member must label their savings account to be one of the following: house, car, vacation
* Investment account:
  + Member must define % of balance allocation
  + Member must define type of investment plan: conservative, moderate, aggressive

For example, if a member’s balance is $1000, then an example balance allocation per account can be:

* Checking account: 70% of balance (e.g. $700 in her/his checking account)
* Savings account: 25% of balance (e.g., $250 in his savings account)
* Investment account: 5% of balance (e.g., $50 in his investment account)

You must also keep track of member’s name, email address, home address, and social-security number. Additionally, you must describe a bank which has all it’s members.

You are required to create code to demonstrate the usage of your classes (7 points).

Class Account {

private Double Balance = 0;

Public Account(Double balance){

Balance = balance;

}

Public double getBalance(){ return Balance; }

}

Class checking : Account{

Private double Balance = 0;

Public savings(Double blanace) : base(balance){

Balance = balance;

}

}

Class savings : Account{

Private double Balance = 0;

Public string Label;

Public savings(string label, Double blanace) : base(balance){

Balance = balance;

Label = label;

}

}

Class investment : Account{

Private double Balance = 0;

Public string InventmentType;

Public investment(string iT, Double blanace) : base(balance){

Balance = balance;

InvestmentType = iT;

}

}

Class member{

Private Double gobalBalance =0;

Public string Name;

Private String Email, Address;

Private int Social-security;

Public List<Account> Membership = new List<Account>();

Public member(string name, string email , string address, int ss, List<Account>[3] membership)

{

Name = name;

Email = email;

Address = address;

Social-security = ss;

Membership = membership;

}

Public double getGlobalBalance()

{

foreach (Account acc in Membership){

globalBalance = gobalBalance + acc.getBalance();

}

Return globalBalance;

}

Public string getPercent(){

String pcent = “”;

foreach (Account acc in Membership){

pcent = pcent + (globalBalance / acc.getBalance()).ToString();

pcent = pcent + “\n”;

}

Return pcent+”%”;

}

Public string getnames(){ return Name;}

}

Class Bank{

Private List<member> Members = new List<member>()

Public Bank(List<members> members){

Members = members;

}

Public string listMembers(){

string names =””;

foreach(member people in Members){

names = names + people.getnames();

}

Return names;

}

}

Public void main(){

List<Account> P1 = new List<Account>();

P1.add(checking p1Ch = new checking(500));

P1.add(savings p1sav = new savings(“house”, 500);

P1.add(investment = new investment(“aggressive”,1000);

List<member> MemberList = new List<member>();

Member person1 = new member(“Allen”, “a@a.com”, “111st”, 1111111, P1)

MemberList.add(person1);

Bank adamsBank = new Bank(MemberList);

Console.log(adamsBank.listMembers());

Console.log(person1.getPercent());

}