NgeeAnn Polytechnic
School of Engineering
Electronic & Computer Engineering Division

Object-Oriented Programming (1OOPG)

Tutorial 1c: Derived class (1 hr)

1a) Given the class People below, derived a **Customer** class from People. Customer class has a private string variable **strMember**. Show the constructor for Customer who needs to initialize strMember and pass a parameter to the base.

```
class People
{
        private string strName;
        public People(string name  )
        {
                strName = name;
        }
}
```

1b) Derived a class **Sp** from Customer. Sp has a private float variable fltDeposit. Show the constructor for Sp who needs to initialize fltDeposit and pass parameters to the base.

2. If the People class in Q1a has a **public** property **Name** and **Password** and the derived class Customer has two arrays shown:

```
public string[] CustNames = {"John", "Marry ", "Nicole","Mee Mee", "Yu"};
private string[] custPswd={ "abc", "123", "lovely","lovemee", "loveyu"};
```

Write a public method **GetPswd**() that **returns** and **sets** the Password for Name. Note, derived class **can** access public property of base class directly. If the name is not found return "name not found". Input parameter: none. Return: string password

3. If the People class in Q1a has a **public** property **Name** and **Password** and the derived class **Sp** has two arrays shown. Sp is derived from Customer which is derived from People.

```
public string[] AdminNames = {"John", "Marry ", "Nicole","Mee Mee", "Yu"};
private string[] AdminPswd={ "abc", "123", "lovely","lovemee", "loveyu"};
```

Write a public method **SetAdmin**() that **sets** the public property Member of Customer to 'Admin' if the Name and Password match AdminNames and AdminPswd respectively. Note, derived class **can** access public property of base class directly. Input parameter: none. Return: none.

4 If the People class in Q1a has a method CalTotal() as shown, write a method with same definition in Customer class to provide a discount, intDisc for itemcost.

```
class People
{
        private float fltTotalCost;
        private int intQty;

        public virtual void CalCost(float itemcost)
        {
                fltTotalCost = fltTotalCost + intQty * itemcost;
        }
}

Class Customer : People
{
        private int intDisc;









}
```

5a Explain the diferences in static variable and instance variable.

5b In the class Customer, declare a static variable NumOfCust. And increase this variable in the loginButton_Click() event and display it on **NumLabel**.

```
class Customer {

 }
```

---

inside Form1

```
private void loginButton_Click(object sender, EventArgs e)
{



}
```