

Systems Development: Object Oriented Programming

(H171 35)

Introducing Polymorphism

Employees Walkthrough

Step 1 – assigning a derived-class reference to a base-class variable

*This walkthrough will use your existing CommissionEmployee and BasePlusCommissionEmployee as completed at the end of week 9 walkthrough 4.*

* Create a new Console Application and add the two classes CommissionEmployee and BasePlusCommissionEmployee that you have already created.
* You will be creating a new test application i.e. Program.cs

In your Program.cs file:

* Create a new (instantiate) CommissionEmployee object
  + CommissionEmployee commissionEmployee = new CommissionEmployee("Sue", "Jones", "22-22-22", 10000.00M, 0.06M);
  + this demonstrates our normal behaviour of assigning a base-class reference to a base-class variable (remember a reference to an area in memory is created when we use the “new” keyword)
* Create a new BasePlusCommissionEmployee object
  + BasePlusCommissionEmployee baseEmployee = new BasePlusCommissionEmployee("Bob", "Lewis", "33-33-33", 5000.00M, 0.04M, 300.00M);
  + this demonstrates our normal behaviour of assigning a derived-class reference to a derived-class variable
* Invoke ToString and Earnings on base-class object using base-class variable
  + Console.WriteLine("{0} {1}: \n\n{2}\n{3}: {4:C}\n",

"Call CommissionEmployee’s ToString and Earnings methods ",

"with base-class reference to base-class object",

commissionEmployee.ToString(),

"earnings ", commissionEmployee.Earnings() );

* Invoke ToString and Earnings on derived-class object using derived-class variable
  + Console.WriteLine("{0} {1}: \n\n{2}\n{3}: {4:C}\n",

"Call BasePlusCommissionEmployee’s ToString and Earnings methods ",

"with derived-class reference to derived-class object",

baseEmployee.ToString(),

"earnings ", baseEmployee.Earnings() );

* Invoke ToString and Earnings on **derived-class object** using **base**-class variable
  + Firstly, we assign the reference to **derived**-class object baseEmployee to a **base**-class CommissionEmployee variable
  + CommissionEmployee commissionEmployee2 = baseEmployee;
  + then we invoke the (derived-class) methods ToString and Earnings
  + Console.WriteLine("{0} {1}: \n\n{2}\n{3}: {4:C}\n",

"Call BasePlusCommissionEmployee’s ToString and Earnings methods ",

"with base-class reference to derived-class object",

commissionEmployee2.ToString(),

"earnings ", commissionEmployee2.Earnings() );

* A base-class variable that contains a reference to a derived-class object and is used to call a virtual method actually calls the overriding derived-class version of the method

N.B. Add Console.ReadKey(); to keep the console open

Your app should generate the following output:

Call CommissionEmployee's ToString and Earnings methods with base class reference to base class object:

commission employee: Sue Jones

social security number: 222-22-2222

gross sales: $10,000.00

commission rate: 0.06

earnings: $600.00

Call BasePlusCommissionEmployee's ToString and Earnings methods with derived class reference to derived class object:

base-salaried commission employee: Bob Lewis

social security number: 333-33-3333

gross sales: $5,000.00

commission rate: 0.04

base salary: $300.00

earnings: $500.00

Call BasePlusCommissionEmployee's ToString and Earnings methods with base class reference to derived class object:

base-salaried commission employee: Bob Lewis

social security number: 333-33-3333

gross sales: $5,000.00

commission rate: 0.04

base salary: $300.00

earnings: $500.00