

Systems Development: Object Oriented Programming

(H171 35)

Case Study: Accounts Payable using an Interface

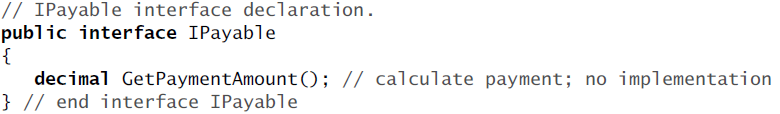
Step 1 – declare Interface IPayable

* IPayable contains method GetPaymentAmount that returns a decimal amount to be paid for any object of any class that implements the interface
* GetPaymentAmount is a general-purpose version of method Earnings of the Employee hierarchy – method Earnings calculates a payment amount specifically for an Employee, while GetPaymentAmount can be applied to a broad range of unrelated objects.

Information Point

When declaring a method in an interface, choose a name that describes the method’s purpose in a general manner, because the method may be implemented by a broad range of unrelated classes.

* Classes Invoice and Employee both represent things for which the company must be able to calculate a payment amount. Both classes implement IPayable, so an app can invoke method GetPaymentAmount on Invoice objects and Employee objects alike. This enables the polymorphic processing of Invoices and Employees required for our Accounts-Payable app.
* Create an IPayable.cs file with the following definition:



* Interface IPayable contains *public abstract* method GetPaymentAmount.
* Interfaces can have any number of members and interface methods can have parameters