

Systems Development: Object Oriented Programming

(H171 35)

Case Study: Accounts Payable using an Interface

Step 3 – modify class Employee to implement Interface IPayable and modify other derived class for use with IPayable

* We now modify class Employee to implement Interface IPayable. We have 2 modifications to make:
  + In the class declaration, indicate that the class implements IPayable i.e. :IPayable
  + Rename method Earnings to be GetPaymentAmount
    - this needs to be **renamed throughout the Employee hierarchy**. As with method Earnings, it does not make sense to implement method GetPaymentAmount in class Employee, because we cannot calculate the earnings payment owed to a general Employee – we need to know the specific type of Employee before we can calculate a payment. When a class implements an Interface it makes a contract stating that the class will either implement each of the methods of the Interface or will declare them abstract - GetPaymentAmount will therefore be an abstract method, and as a result class Employee will have to be declared abstract
* We now modify all derived classes to implement GetPaymentAmount, which is effectively the implementation of Earnings, therefore we change the method name from Earnings to GetPaymentAmount for each derived class –SalariedEmployee, HourlyEmployee, CommissionEmployee, and BasePlusCommissionEmployee