Q1)(Employee Class) Create a class called **Employee** that includes three pieces of information as either instance variables or automatic properties—a first name (type string), a last name (type string) and a monthly salary (decimal).

Your class should have a constructor that initializes the three values. Provide a property with a get and set accessor for any instance variables.

If the monthly salary is negative, the set accessor should leave the instance variable unchanged. Write a test app named EmployeeTest that demonstrates class Employee's capabilities. Create two Employee objects and display each object's yearly salary. Then give each Employee a 10% raise and display each Employee's yearly salary again.

Q2)(Date Class) Create a class called Date that includes three pieces of information as automatic properties—a month (type int), a day (type int) and a year (type int). Your class should have a constructor that initializes the three automatic properties and assumes that the values provided are correct. Provide a method DisplayDate that displays the month, day and year separated by forward slashes (/). Write a test app named DateTest that demonstrates class Date's capabilities.

Q3) (Invoice Class) Create a class called Invoice that a hardware store might use to represent an invoice for an item sold at the store. An Invoice should include four pieces of information as either instance variables or automatic properties—a part number (type string), a part description (type string), a quantity of the item being purchased (type int) and a price per item (decimal). Your class should have a constructor that initializes the four values. Provide a property with a get and set accessor for any instance variables. For the Quantity and PricePerItem properties, if the value passed to the set accessor is negative, the value of the instance variable should be left unchanged. Also, provide a method named GetInvoiceAmount that calculates the invoice amount (i.e., multiplies the quantity by the price per item), then returns the amount as a decimal value. Write a test app named InvoiceTest that demonstrates class Invoice's capabilities