Introduction to Classes Exercises

Introduction to Classes contains a series of exercises which require you to define and use classes of increasing difficulty. The series is grouped into three sets: Easy, Medium, and Difficult.

A starter Visual Studio Solution containing all class and test source files you'll need has been created for you.

Easier

Product

Class Properties

Property Name	Data Type	Get	Set	Private Field Name	Description
Name	string	X	X	name	Holds the name of the product.
Price	decimal	X	X	price	Holds the price of the product.
WeightInOunces	double	X	X	weightInOunces	Holds the weight (in ounces) of the product.

Company

Class Properties

Property Name	Data Type	Get	Set	Private Field Name	Description
Name	string	X		name	Holds the name of the company.
NumberOfEmployees	int	X	X	numberOfEmployees	Holds the number of employees.
Revenue	decimal	X	X	revenue	Holds the company revenue.
Expenses	decimal	X	X	expenses	Holds the company expenses.

Constructors

Signature	Description
Company(string startingName)	Starting name of the company. This should set the value of the name field.

Methods

Method Name	Return Type	Description
GetCompanySize()	string	A company is "small" if less than 50 employees, "medium" if between 50 and 250 employees, "large" if greater than 250 employees
GetProfit()	decimal	Calculated by subtracting expenses from revenue.

Person

Class Properties

Property Name	Data Type	Get	Set	Private Field Name	Description
FirstName	string	X	X	firstName	Holds the first name of the person.
LastName	string	X	X	lastName	Holds the last name of the person.
Age	int	X	X	age	Holds the age of the person.

Methods

Method Name	Return Type	Description
GetFullName()	string	Returns the First Name + Last Name of the Person.
IsAdult()	bool	Returns true if the person is 18 or older.

Medium Difficulty

Dog

Class Properties

Property Name	Data Type	Get	Set	Private Field Name	Description
IsSleeping	bool	X		isSleeping	TRUE if the dog is asleep. FALSE if not. **All new dogs are awake by default**

Constructors

Signature	Description
Dog()	Dog constructor takes no arguments. **All new dogs are awake by default**

Methods

Method Name	Return Type	Description
MakeSound()	string	Returns "Zzzzz..." if the dog is asleep. Returns "woof!" if the dog is awake.
Sleep()	void	Sets isSleeping to true.
WakeUp()	void	Sets isSleeping to false.

Shopping Cart

Class Properties

Property Name	Data Type	Get	Set	Private Field Name	Description
TotalNumberOfItems	int	X		totalNumberOfItems	The number of items in the shopping cart. **All shopping carts have 0 items by default**
TotalAmountOwed	decimal	X		totalAmountOwed	The total for the shopping cart. **All shopping carts have 0.0 owed by default**

Methods

Method Name	Return Type	Description
GetAveragePricePerItem()	decimal	Returns the totalAmountOwed / totalNumberOfItems.
AddItems(int numberOfItems, decimal pricePerItem)	void	Updates totalNumberOfItems and increases totalAmountOwed by (pricePerItem * numberOfItems)
Empty()	void	Returns totalNumberOfItems and totalAmountOwed to 0.

Difficult

Calculator

Class Properties

Property Name	Data Type	Get	Set	Private Field Name	Description
Result	int	X		result	Holds the current value of the calculator

Constructors

Signature	Description
Calculator(int startingResult)	Starting value of the calculator. Initialize result to the value of startingResult

Methods

Method Name	Return Type	Description
Add(int addend)	int	Adds addend to result and returns the current value of result.
Subtract(int subtrahend)	int	Subtracts subtrahend from the current value of result and returns the current value of result.
Multipy(int multiplier)	int	Multiplies current result by multiplier and returns the current value of result.
Power(int exponent)	int	Raises result to power of exponent. Negative exponents should use the absolute value. Returns the current value of result
Reset()	void	Resets result to 0.