```
public class Validator
{

	// TODO - CODE THE VALIDATOR
	// add the method that will validate all the input
	// collected from the form

	// static variable to hold any error that occurs
	private static String errorMessage = "";

	public static String getOrderID(String anID)
	{
		// run the validation
		// add any errors to the error message
		if (anID.length() == 0)
		{
			errorMessage += "Order ID cannot be blank\n";
		}
		return anID;
	}

	public static String getFirstName(String aFirstName)
	{
		if (aFirstName.length() == 0)
		{
			errorMessage += "First Name cannot be blank\n";
		}
		return aFirstName;
	}

	public static String getLastName(String aLastName)
	{
		if (aLastName.length() == 0)
		{
			errorMessage += "Last Name cannot be blank\n";
		}
		return aLastName;
	}

	public static String getMemberShip(String aMemberShip)
	{
		if(aMemberShip.length() == 0)
		{
			errorMessage += "MemberShip cannot be blank\n";
		}
		if(aMemberShip.trim().matches("[a-zA-Z]{2}\\d{4}") == false)
		{
			errorMessage += "MemberShip is in invalid Format!\n";
		}
		return aMemberShip;
	}

	public static String getPhone(String aPhone)
	{
		if(aPhone.length() == 0)
		{
			errorMessage += "Phone cannot be blank\n";
		}
		if(aPhone.trim().matches("[(]?(\\d{3})[-)\\s]?(\\d{3})[-\\s]?(\\d{4})") == fal…
		{
			errorMessage += "Phone in inproper format\n";
		}
		return aPhone;
```

```
}

public static String getStreet(String aStreet)
{
        if(aStreet.length() == 0)
        {
                errorMessage += "Street cannot be blank\n";
        }
        return aStreet;
}

public static String getCity(String aCity)
{
        if(aCity.length() == 0)
        {
                errorMessage += "City cannot be blank\n";
        }
        return aCity;
}

public static String getState(String aState)
{
        if(aState.length() == 0)
        {
                errorMessage += "State cannot be blank\n";
        }
        if(aState.trim().matches("[a-zA-Z]{2}") == false)
        {
                errorMessage += "Not a valid State!\n";
        }
        return aState;
}

public static String getZip(String aZip)
{
        if(aZip.length() == 0)
        {
                errorMessage += "Zip cannot be blank\n";
        }
        if(aZip.trim().matches("\\d{5}") == false)
        {
                errorMessage += "Not a valid zip\n";
        }
        return aZip;
}

public static String getQty(String aQty)
{
        if(aQty.matches("\\d+?") == false)
        {
                errorMessage += "Quantity is not a vaild whole number\n";
        }
        return aQty;
}

public static String getPrice(String aPrice)
{
        if(aPrice.matches("(\\$)?\\d+(\\.\\d*)?") == false)
        {
                errorMessage += "Price is not valid\n";
        }
        return aPrice;
}
```

```
        // method to return the error message
        public static String getError()
        {
                return errorMessage;
        }

        // method to clear the error message
        public static void clearError()
        {
                errorMessage = "";
        }

        //method to add an error message
        public static void setError(String anError)
        {
                errorMessage += anError;
        }

}
```