What is the diff bw APEX:MESSAGE and MESSAGES IN VISUALFORCE page?

**<APEX:MESSAGE>**

It is used to display an error on only a very specific field. It is used to allow the developer to place field specific errors in a specific location.

**Example:**

<apex:page controller="TestMessageController">

<apex:form >

<apex:outputLabel value="Test String" /><apex:inputText value="{!test.Name}" id="test"/>

<br/><apex:message for="test"/>

</apex:form>

**controller:**

public class TestMessageController{

public Account test{get;set;}

public TestMessageController(){

test = new Account();

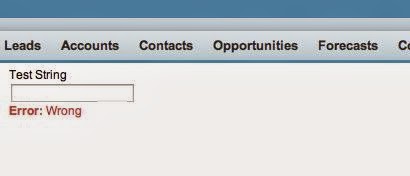
test.Id.addError('Correct');

test.Name.addError('Wrong');

}

}

**VF Page:**

[](http://2.bp.blogspot.com/-SaUX-GOgJm0/Uk7USaQOm9I/AAAAAAAABBU/jdP3JU2W8WY/s1600/Visualforce_message.JPG)

**<APEX:MESSAGES>**

It is similar to apex:message, but it displays all of the errors. These errors are displayed as a list with no styling. Let's take the last example we used in the apex:message and simply add the apex:messages tag to the Visualforce.

This provides no formatting for the errors, however, it does display them all. The use case for using apex:messages is to display all errors on the page while providing your own styling.

**Example:**

<apex:page controller="TestMessageController">

<apex:pageMessages />

<apex:pageMessage summary="This is apex:message" severity="info" strength="2"/>

<apex:messages />

<apex:form >

<apex:outputLabel value="Test String" /><apex:inputField value="{!test.Name}" id="test"/>

<br/><apex:message for="test"/>

</apex:form>

</apex:page>

**Controller:**

public class TestMessageController{

public Account test{get;set;}

public TestMessageController(){

ApexPages.addMessage(new ApexPages.Message(ApexPages.Severity.ERROR, 'This is apex:pageMessages'));

test = new Account();

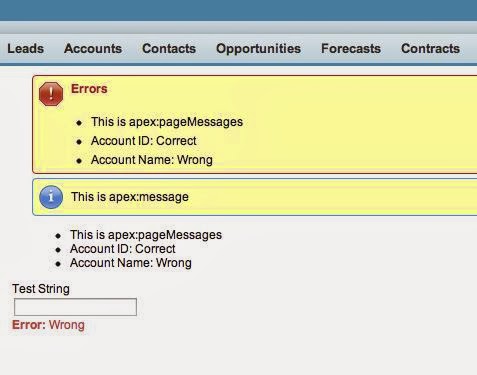
test.Id.addError('Correct');

test.Name.addError('Wrong');

}

}

**VF Page:**

[](http://3.bp.blogspot.com/-MRmnH1-gOZs/Uk7Xcm7ZhwI/AAAAAAAABBg/L2FrRz2WdJ8/s1600/Visualforce_messages.JPG)

apex:messages shows all of the errors with no formatting, and apex:message shows only the error for the specified field.

Let me start off by explaining each of these in more detail.

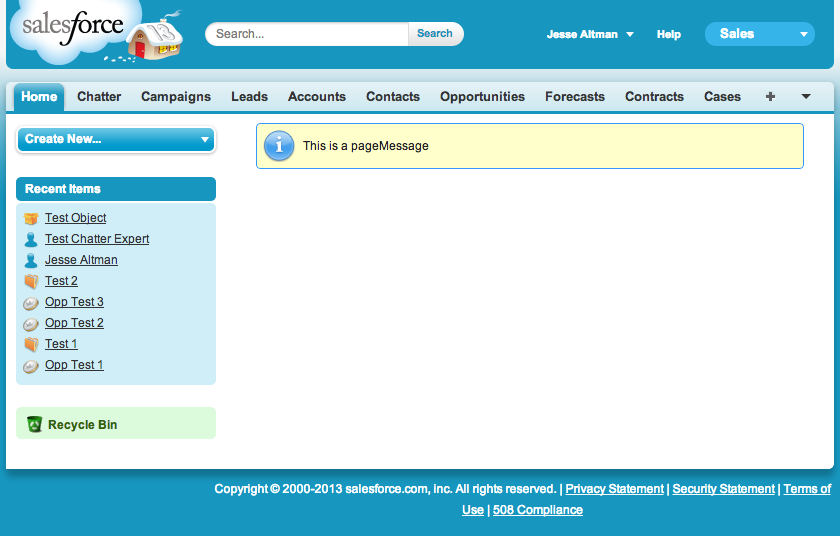
apex:pageMessage is used to display a single custom message using the Salesforce formatting. You can specify the severity (which will control the display of the message box) and the strength (which will control the size of the box). Take the following code for example:

<apex:page >

<apex:pageMessage summary="This is a pageMessage" severity="info" strength="3"/>

</apex:page>

Renders as:



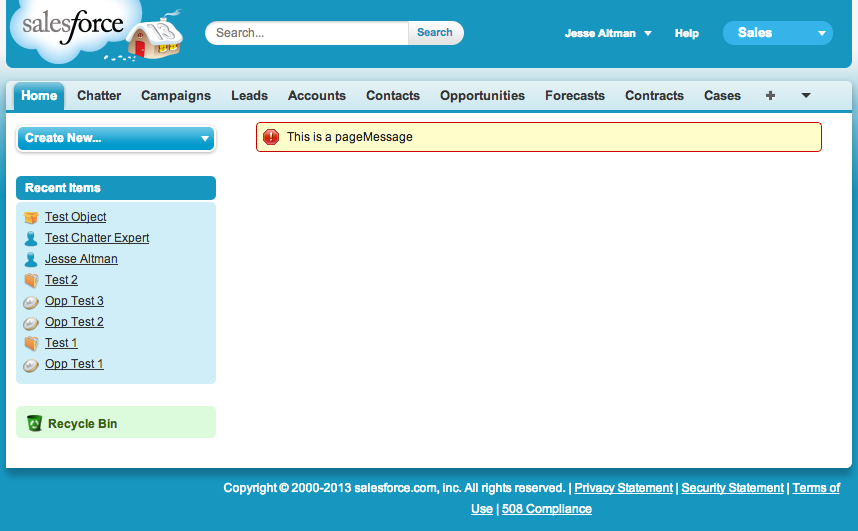
Now let's tweak that code slightly:

<apex:page >

<apex:pageMessage summary="This is a pageMessage" severity="error" strength="1"/>

</apex:page>

Notice how I changed the severity and the strength. This controls the size and the display of the message. This renders as:



This can be used to display any custom message that you always want to appear on the screen. There may be a case that whenever a user is on a form you may want a warning to appear to display important information to them.

apex:pageMessages is used to display all of the messages on a page. It will display Salesforce generated messages as well as custom messages added to the ApexPages class. Let's take a look at it in use:

<apex:page controller="TestMessageController">

<apex:pageMessages />

</apex:page>

public class TestMessageController{

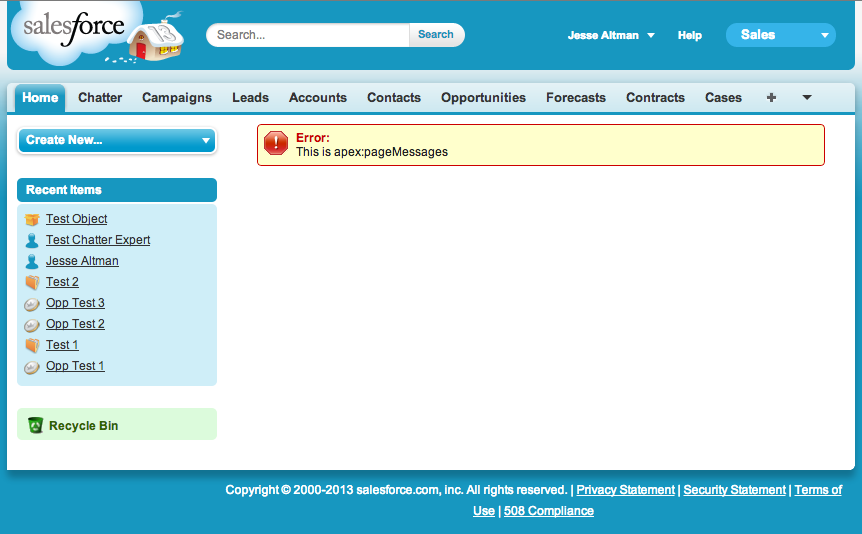
public TestMessageController(){

ApexPages.addMessage(new ApexPages.Message(ApexPages.Severity.ERROR, 'This is apex:pageMessages'));

}

}

This renders as:



Let's add some more messages and see how it reacts:

<apex:page controller="TestMessageController">

<apex:pageMessages />

</apex:page>

public class TestMessageController{

public TestMessageController(){

ApexPages.addMessage(new ApexPages.Message(ApexPages.Severity.ERROR, 'This is apex:pageMessages'));

ApexPages.addMessage(new ApexPages.Message(ApexPages.Severity.ERROR, 'This is apex:pageMessages still'));

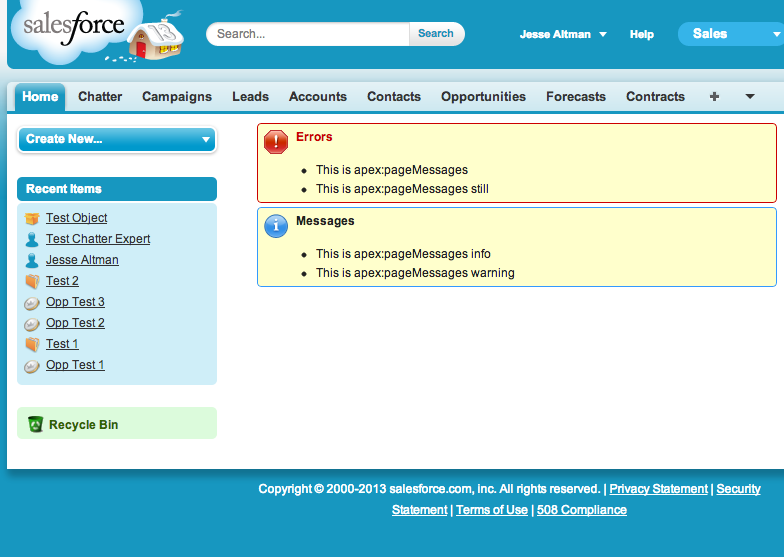
ApexPages.addMessage(new ApexPages.Message(ApexPages.Severity.INFO, 'This is apex:pageMessages info'));

ApexPages.addMessage(new ApexPages.Message(ApexPages.Severity.INFO, 'This is apex:pageMessages warning'));

}

}

This renders as:



Notice how that single instance of the apex:pageMessages handles all of the errors on the page. The use case for this object is to capture all errors on the page. It should be used on most custom Visualforce page as a catch all to let users know what errors occur (if no other messaging is used).

apex:message is used to display an error on only a very specific field. It is used to allow the developer to place field specific errors in a specific location. Take the following code for example:

<apex:page controller="TestMessageController">

<apex:form >

<apex:outputLabel value="Test String" /><apex:inputText value="{!test.Name}" id="test"/>

<br/><apex:message for="test"/>

</apex:form>

public class TestMessageController{

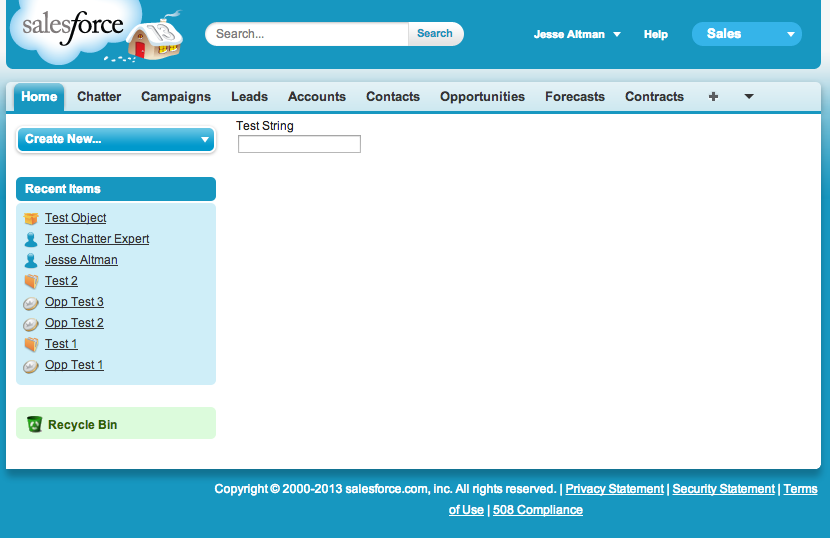
public Account test{get;set;}

public TestMessageController(){

}

}

renders as:



Notice how this currently doesn't have a message displayed. Also notice how this particular element is specified specifically for one field. The error will only occur on that one field. Let's change the code a bit and take a look at this with an error on the Name field:

<apex:page controller="TestMessageController">

<apex:form >

<apex:outputLabel value="Test String" /><apex:inputField value="{!test.Name}" id="test"/>

<br/><apex:message for="test"/>

</apex:form>

</apex:page>

public class TestMessageController{

public Account test{get;set;}

public TestMessageController(){

test = new Account();

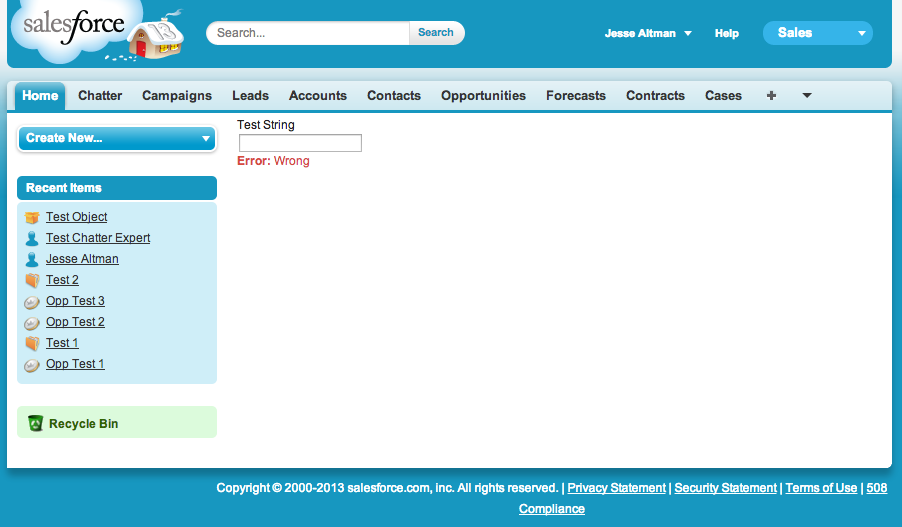
test.Id.addError('Correct');

test.Name.addError('Wrong');

}

}

This renders as:



Notice how only the Name error is shown because that is the only field that has an apex:message associated to it. Also notice that although this does have some styling, the styling is minimalistic and Salesforce expects the developer to handle the styling as necessary in this situation.

The final thing to look at is apex:messages. apex:messages is similar to apex:message, but it displays all of the errors. These errors are displayed as a list with no styling. Let's take the last example we used in the apex:message and simply add the apex:messages tag to the Visualforce. The code:

<apex:page controller="TestMessageController">

<apex:messages />

<apex:form >

<apex:outputLabel value="Test String" /><apex:inputField value="{!test.Name}" id="test"/>

<br/><apex:message for="test"/>

</apex:form>

</apex:page>

public class TestMessageController{

public Account test{get;set;}

public TestMessageController(){

test = new Account();

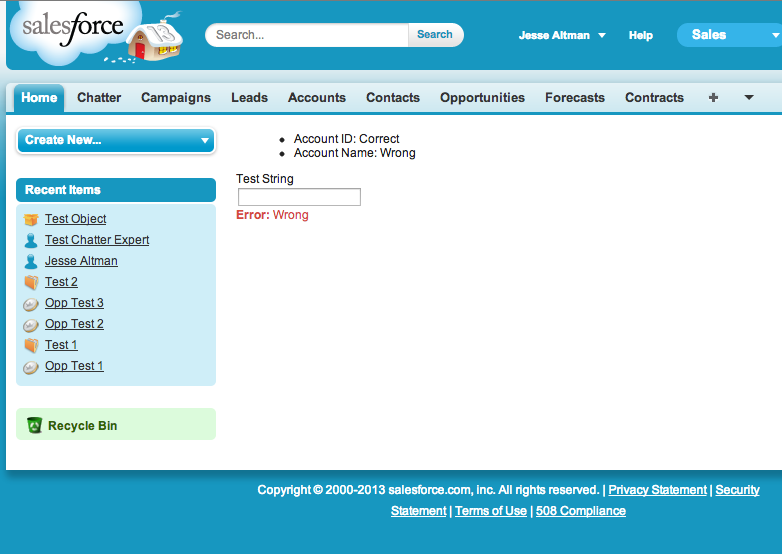
test.Id.addError('Correct');

test.Name.addError('Wrong');

}

}

renders as:



This provides no formatting for the errors, however, it does display them all. The use case for using apex:messages is to display all errors on the page while providing your own styling.

Now, to close it all out, let's take a look at them all together. The code:

<apex:page controller="TestMessageController">

<apex:pageMessages />

<apex:pageMessage summary="This is apex:message" severity="info" strength="2"/>

<apex:messages />

<apex:form >

<apex:outputLabel value="Test String" /><apex:inputField value="{!test.Name}" id="test"/>

<br/><apex:message for="test"/>

</apex:form>

</apex:page>

public class TestMessageController{

public Account test{get;set;}

public TestMessageController(){

ApexPages.addMessage(new ApexPages.Message(ApexPages.Severity.ERROR, 'This is apex:pageMessages'));

test = new Account();

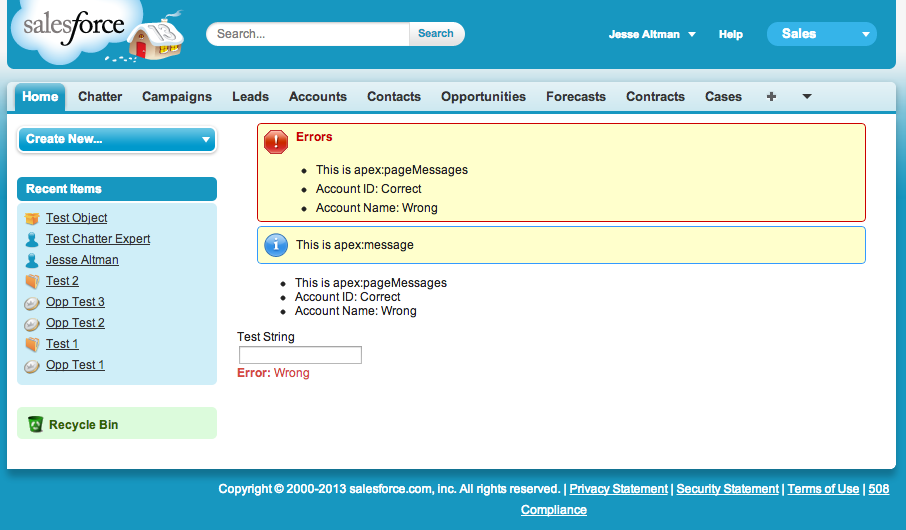
test.Id.addError('Correct');

test.Name.addError('Wrong');

}

}

renders as:



As expected, apex:pageMessages shows all of the errors with formatting, apex:pageMessage only shows the specified message, apex:messages shows all of the errors with no formatting, and apex:message shows only the error for the specified field.