

Binding Data and Handling a User Event

Create an ArrayCollection variable

1. Open **AdobeODT.mxml** in the **AdobeODT** project.
2. After the beginning **Application** tag, create an **Array** using MXML.

```
<mx:Array>  
</mx:Array>
```

3. Add an **id** property with the value **roomList**.

```
<mx:Array id="roomList">  
</mx:Array>
```

4. Populate the array using the following values. Surround each value with the **String** class.

```
Colorado Room  
Denver Room  
Mile High Room  
Greeley Room
```

```
<mx:Array id="roomList">  
  <mx:String>Colorado Room</mx:String>  
  <mx:String>Denver Room</mx:String>  
  <mx:String>Mile High Room</mx:String>  
  <mx:String>Greeley Room</mx:String>  
</mx:Array>
```

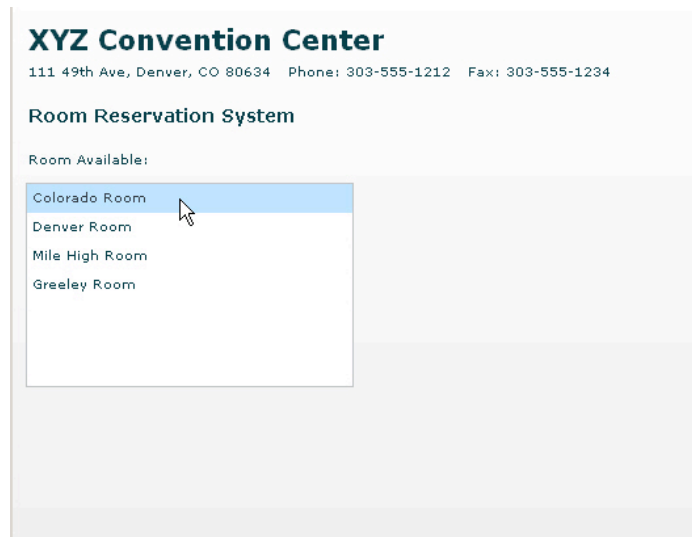
Use the variable as the dataProvider for the List control

1. In the **Outline** view, locate the **List** control.
2. Add a `dataProvider` property and bind `roomList` using curly braces.

```
<mx:List id="dg" x="10" y="134" width="250"  
        dataProvider="{roomList}" />
```

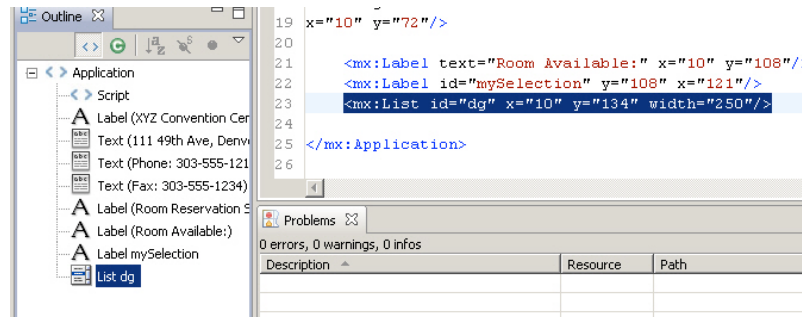
3. Save the file and run.

You should see the **List** control populated with the values from the **Array**.



Create a user event

1. Using the **Outline** view locate the **List** control.



2. Add an `itemClick` event that is handled by a `changeHandler` method. Pass `event` as a parameter to the method.

```
<mx:List id="dg" x="10" y="134" width="250"
  dataProvider="{roomList}"
  itemClick="changeHandler(event)"/>
```

Create the changeHandler listener

1. After the beginning **Application** tag, add a **Script** block: `<mx:Script>`. Flex Builder will complete the code for you when you enter the closing bracket (`>`) of the **Script** tag.
2. Within the **Script** block after the last import statement, import the `ListEvent` class.

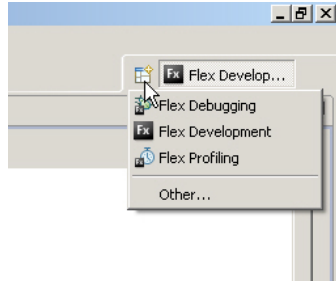
```
<mx:Script>
  <![CDATA[
    import mx.events.ListEvent;
  ]]>
</mx:Script>
```

3. After the import statement, create a `private` function named `changeHandler` that takes one parameter named `event` datatyped as `ListEvent`. The function returns `void`.

```
private function changeHandler(event:ListEvent):void{
}

```

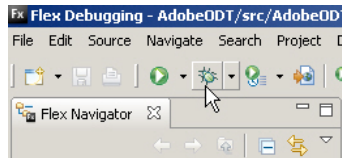
4. Switch to the **Flex Debugging** perspective.



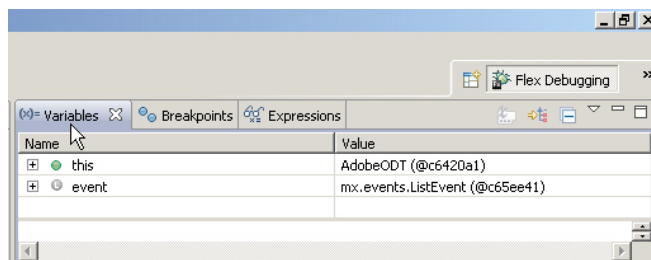
5. Add a **break point** to the line with the closing curly brace by double clicking in the marker bar next to the line number associated with the brace.

```
9      import mx.events.  
10     private function  
11     }  
12     ]]>  
13     </mx:Script>
```

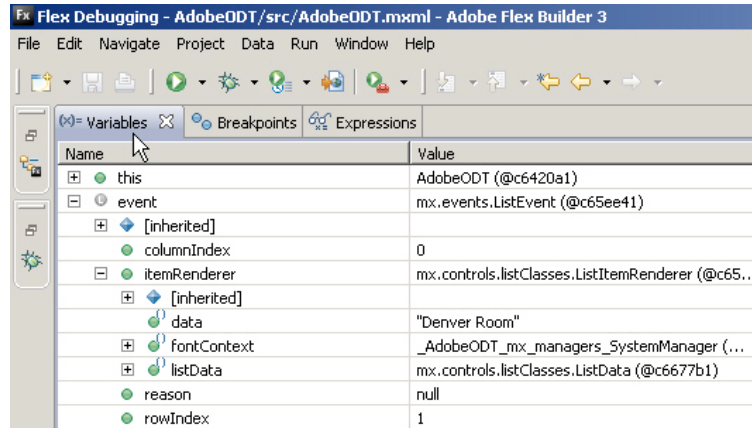
6. Save and debug the application.



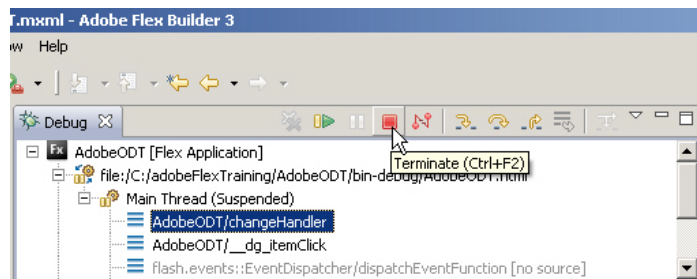
7. In the browser, click a room in the list. This will return you to **Flex Builder**.
8. Double click the **Variables** view tab to maximize it.



9. Expand the **event** variable, and then expand the **itemRenderer** variable. You should see the **data** variable with the value you selected.



10. Double click the **Variables** view to collapse it.
11. Stop the debugging session by clicking the red square.



12. Return to the **Flex Development** perspective.