



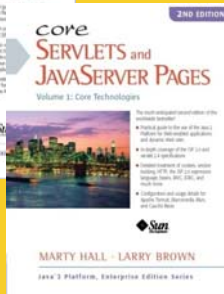
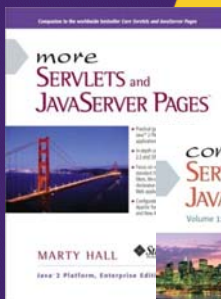
PrimeFaces: Charts and Graphs

Originals of slides and source code for examples: <http://www.coreservlets.com/JSF-Tutorial/primefaces/>
Also see the JSF 2 tutorial – <http://www.coreservlets.com/JSF-Tutorial/jsf2/>
and customized JSF2 and PrimeFaces training courses – <http://courses.coreservlets.com/jsf-training.html>



Customized Java EE Training: <http://courses.coreservlets.com/>

Java 7, Java 8, JSF 2, PrimeFaces, Android, JSP, Ajax, jQuery, Spring MVC, RESTful Web Services, GWT, Hadoop
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.



For live training on JSF 2, PrimeFaces, or other Java EE topics, email hall@coreservlets.com
Marty is also available for consulting and development support

Taught by the author of *Core Servlets and JSP*, this tutorial, and JSF 2.2 version of *Core JSF*. Available at public venues, or customized versions can be held on-site at your organization.

- Courses developed and taught by Marty Hall
 - JSF 2, PrimeFaces, Ajax, jQuery, Spring MVC, JSP, Android, general Java, Java 8 lambdas/streams, GWT, custom topic mix
 - Courses available in any location worldwide. Maryland/DC area companies can also choose afternoon/evening courses.
- Courses developed and taught by coreservlets.com experts (edited by Marty)
 - Hadoop, Spring, Hibernate/JPA, RESTful Web Services

Contact hall@coreservlets.com for details



Topics in This Section

- Overview
- Pie charts
- Line charts
- Area charts (filled line charts)
- Bar charts
- Interactive (Ajax-enabled) charts

5

© 2015 [Marty Hall](#)



Overview



Customized Java EE Training: <http://courses.coreservlets.com/>

Java 7, Java 8, JSF 2, PrimeFaces, Android, JSP, Ajax, jQuery, Spring MVC, RESTful Web Services, GWT, Hadoop

Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

Overview

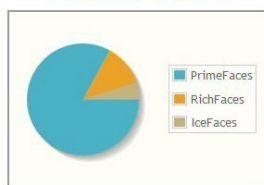
- **Uses the jqplot library under the hood**
 - An open-source plugin for jQuery
 - <http://www.jqplot.com/>
- **Uses HTML 5 Canvas to draw**
 - Works on all recent versions of Chrome, Firefox, Safari, and Opera
 - Works natively in IE 9 and later
 - IE 7 and 8 fall back to the excanvas library
- **Wrapped up as a JSF component**
 - Data is defined on the server
 - Ajax support for interactively selecting graph points, bars, or pie wedges

7

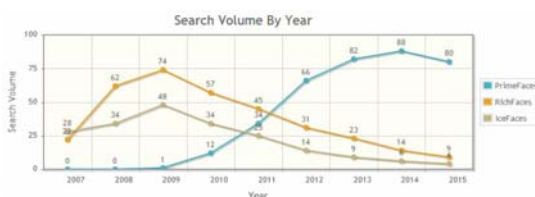
Basic Charts (Covered in this Tutorial)

Pie chart

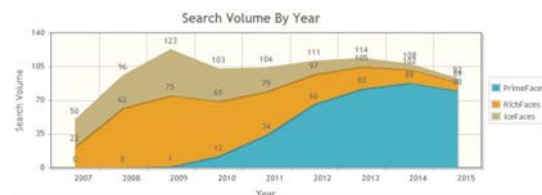
2015 Search Volume



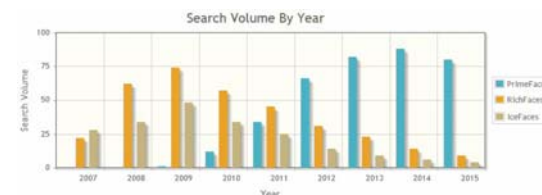
Line chart



Area chart



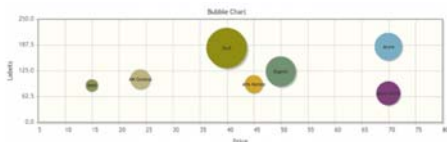
Bar chart



8

Other Chart Types (See PrimeFaces User's Guide)

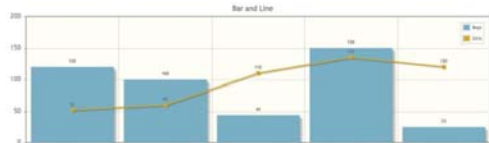
Bubble Chart



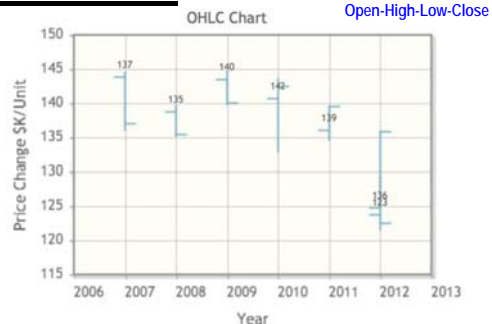
MeterGauge Chart



Combined Chart



OHLC Chart



Donut Chart



9

© 2015 Marty Hall



Pie Charts



Customized Java EE Training: <http://courses.coreservlets.com/>

Java 7, Java 8, JSF 2, PrimeFaces, Android, JSP, Ajax, jQuery, Spring MVC, RESTful Web Services, GWT, Hadoop
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

Overview

- **Java**
 - Make PieChartModel
 - Call **set** with labels and values, once for each pie wedge
 - Call setTitle with overall chart title
 - Call setLegendPosition with legend location
 - n, ne, e, se, s, sw, w, nw
- **HTML (p:chart)**
 - type
 - "pie"
 - model
 - The PieChartModel from above
 - style or styleClass
 - To control size/position of the pie chart

11

Syntax Summary: HTML

```
<p:chart type="pie"  
        model="#{someBean.model}"  
        styleClass="optional-css-name"/>
```

12

Syntax Summary: Java

```
private void initializeModel() {
    model = new PieChartModel();
    model.set("Wedge label 1", wedgeValue1);
    model.set("Wedge label 2", wedgeValue2);
    ...
    model.setTitle("Overall chart title");
    model.setLegendPosition(one-of-e-w-n-s-etc);
}

public PieChartModel getModel() {
    return(model);
}
```

13

Example: HTML

```
<p:fieldset legend="Pie Chart">
<h:panelGrid columns="3">
<p:chart type="pie" model="#{pieChart.model2011}"
    styleClass="medium-pie"/>
<p:chart type="pie" model="#{pieChart.model2013}"
    styleClass="medium-pie"/>
<p:chart type="pie" model="#{pieChart.model2015}"
    styleClass="medium-pie"/>
</h:panelGrid>
</p:fieldset>
```

```
CSS File:
.medium-pie {
    width: 300px;
    height: 250px;
}
```

14

Example: Java (Part 1 – Setup)

```
@ManagedBean
@ApplicationScoped // Data does not change
public class PieChart {
    private PieChartModel model2011, model2013,
                          model2015;

    public PieChart() {
        initialize2011();
        initialize2013();
        initialize2015();
    }
}
```

15

Example: Java (Part 2 – First Pie)

```
private void initialize2011() {
    model2011 = new PieChartModel();
    model2011.set("PrimeFaces", 26);
    model2011.set("RichFaces", 48);
    model2011.set("IceFaces", 27);
    model2011.setTitle("2011 Search Volume");
    model2011.setLegendPosition("e");
}

public PieChartModel getModel2011() {
    return(model2011);
}
```

16

Example: Java (Part 3 – Second Pie)

```
private void initialize2013() {
    model2013 = new PieChartModel();
    model2013.set("PrimeFaces", 65);
    model2013.set("RichFaces", 22);
    model2013.set("IceFaces", 11);
    model2013.setTitle("2013 Search Volume");
    model2013.setLegendPosition("e");
}

public PieChartModel getModel2013() {
    return(model2013);
}
```

17

Example: Java (Part 4 – Third Pie)

```
private void initialize2015() {
    model2015 = new PieChartModel();
    model2015.set("PrimeFaces", 68);
    model2015.set("RichFaces", 10);
    model2015.set("IceFaces", 4);
    model2015.setTitle("2015 Search Volume");
    model2015.setLegendPosition("e");
}

public PieChartModel getModel2015() {
    return(model2015);
}
}
```

18

Example: Results



19

© 2015 Marty Hall



Line Charts (Line Graphs)



Customized Java EE Training: <http://courses.coreservlets.com/>

Java 7, Java 8, JSF 2, PrimeFaces, Android, JSP, Ajax, jQuery, Spring MVC, RESTful Web Services, GWT, Hadoop
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

Overview

- **Java**
 - Make LineChartSeries
 - Label entire line with getLabel
 - Specify data points with set(xValue, yValue). The x values can be any strings: they just become consecutive labels on the x axis. The y values must be numbers.
 - Make LineChartModel
 - Set axis information and legend position
 - Call **addSeries** for each LineChartSeries from above
- **HTML (p:chart)**
 - type: "line"
 - model: the LineChartModel from above

21

Syntax Summary: HTML

```
<p:chart type="line"  
        model="#{someBean.model}"  
        styleClass="optional-css-name"/>
```

22

Syntax Summary: Java (Part 1 – LineChartSeries)

```
private void initializeSeries1() {
    series1 = new LineChartSeries();
    series1.setLabel("Series 1");
    series1.set("Q1", 2);
    series1.set("Q2", 3);
    ...
}
```

```
private void initializeSeries1() {
    series2 = new LineChartSeries();
    series2.setLabel("Series 1");
    series2.set("Q1", 5);
    series1.set("Q2", 3);
    ...
}
```

The first argument to set (the x values) can be numbers or strings.

If they are numbers, you should not use a category axis, and the system will make an evenly spaced X axis even if your entries are not evenly spaced.

If the x values are strings, you should use a category axis, and the exact x values listed will be category labels.

23

Syntax Summary: Java (Part 2 – LineChartModel)

```
private void initializeModel() {
    model = new LineChartModel();
    setAxisInfo();
    model.addSeries(series1);
    model.addSeries(series2);
    ...
}

public LineChartModel getModel() {
    return(model);
}
```

24

Syntax Summary: Java (Part 3 – Axis Information)

```
private void setAxisInfo() {
    model.setTitle("Overall Graph Title");
    model.setLegendPosition("e"); // Or w, s, etc.
    model.setShowPointLabels(true);
    model.getAxes()
        .put(AxisType.X,
            new CategoryAxis("x-axis-title"));
    Axis yAxis = model.getAxis(AxisType.Y);
    yAxis.setMin(minValue);
    yAxis.setMax(maxValue);
    yAxis.setLabel("y-axis-title");
    ...
}
```

Omit this if using numbers (rather than Strings) for the x values. If using a numeric x axis (rather than a category axis), see also `setTickInterval` at <http://www.primefaces.org/docs/api/5.1/org/primefaces/model/chart/Axis.html>

25

Example: HTML

```
<p:fieldset legend="Line Chart">
  <p:chart type="line"
    model="#{lineChart.yearModel}"/>
</p:fieldset>
```

26

Example: Java (Part 1A – Chart Data Part 1)

```
public class ChartData {
    public static final int START_YEAR = 2007;
    private static final int[] primeFacesNumbers =
        { 0, 0, 1, 12, 34, 66, 82, 88, 80 };
    private static final int[] richFacesNumbers =
        { 22, 62, 74, 57, 45, 31, 23, 14, 9 };
    private static final int[] iceFacesNumbers =
        { 28, 34, 48, 34, 25, 14, 9, 6, 4 };

    private LineChartSeries primeFacesSeries;
    private LineChartSeries richFacesSeries;
    private LineChartSeries iceFacesSeries;

    public ChartData() {
        buildLineChartSeries();
    }
}
```

27

This is in a separate class because the next graphs use the same series data.

Example: Java (Part 1B – Chart Data Part 2)

```
private void buildLineChartSeries() {
    primeFacesSeries = new LineChartSeries();
    primeFacesSeries.setLabel("PrimeFaces");
    richFacesSeries = new LineChartSeries();
    richFacesSeries.setLabel("RichFaces");
    iceFacesSeries = new LineChartSeries();
    iceFacesSeries.setLabel("IceFaces");
    int numYears = primeFacesNumbers.length;
    for(int i=0; i<numYears; i++) {
        String xValue = String.valueOf(i + START_YEAR);
        int yValue = primeFacesNumbers[i];
        primeFacesSeries.set(xValue, yValue);
        yValue = richFacesNumbers[i];
        richFacesSeries.set(xValue, yValue);
        yValue = iceFacesNumbers[i];
        iceFacesSeries.set(xValue, yValue);
    }
}
```

28

Notice that the x values are strings ("2007", "2008", etc.)

Example: Java (Part 1C – Chart Data Part 3)

```
public LineChartSeries primeFacesSeries() {
    return(primeFacesSeries);
}

public LineChartSeries richFacesSeries() {
    return(richFacesSeries);
}

public LineChartSeries iceFacesSeries() {
    return(iceFacesSeries);
}
```

29

Example: Java (Part 2A – Model Main Settings)

```
@ManagedBean
@ApplicationScoped
public class LineChart {
    private LineChartModel yearModel;

    public LineChart() {
        yearModel = new LineChartModel();
        setAxisInfo();
        ChartData sampleData = new ChartData();
        LineChartSeries primeFacesSeries = sampleData.primeFacesSeries();
        yearModel.addSeries(primeFacesSeries);
        LineChartSeries richFacesSeries = sampleData.richFacesSeries();
        yearModel.addSeries(richFacesSeries);
        LineChartSeries iceFacesSeries = sampleData.iceFacesSeries();
        yearModel.addSeries(iceFacesSeries);
    }

    public LineChartModel getYearModel() {
        return(yearModel);
    }
}
```

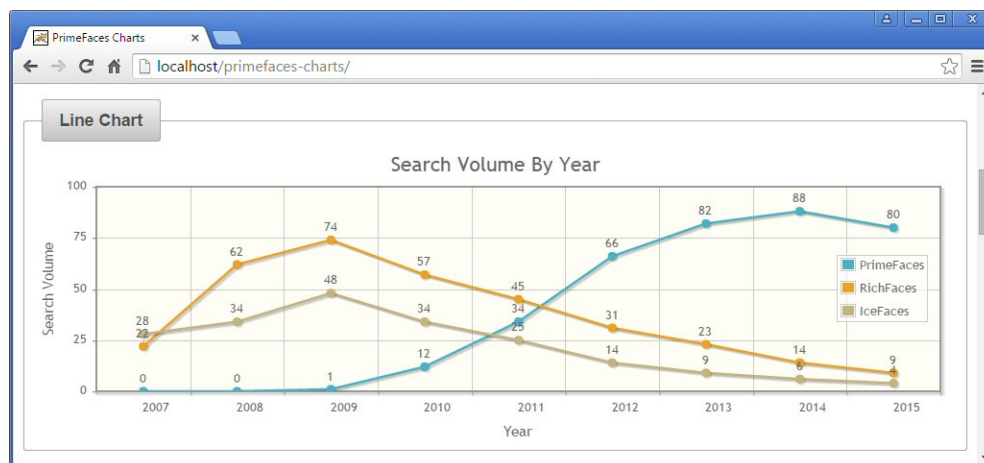
30

Example: Java (Part 2B – Model Axis Settings)

```
private void setAxisInfo() {  
    yearModel.setTitle("Search Volume By Year");  
    yearModel.setLegendPosition("e");  
    yearModel.setShowPointLabels(true);  
    yearModel.getAxes().put(AxisType.X, new CategoryAxis("Year"));  
    Axis yAxis = yearModel.getAxis(AxisType.Y);  
    yAxis.setMin(0);  
    yAxis.setMax(100);  
    yAxis.setLabel("Search Volume");  
}  
}
```

31

Example: Results



32



Area Charts (Filled/Stacked Line Graphs)



Customized Java EE Training: <http://courses.coreservlets.com/>

Java 7, Java 8, JSF 2, PrimeFaces, Android, JSP, Ajax, jQuery, Spring MVC, RESTful Web Services, GWT, Hadoop
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

Overview

- **Java**
 - Make LineChartSeries
 - Same as last example
 - Make LineChartModel
 - Set axis information and legend position
 - Same as last example
 - Call **setFill(true)** on each LineChartSeries from above
 - Call addSeries for each LineChartSeries from above
 - Same as last example
 - Call **setStacked(true)** on the model
- **HTML (p:chart)**
 - type: "line"
 - model: the LineChartModel from above
 - Same as last example

Syntax Summary: HTML

```
<p:chart type="line"  
        model="#{someBean.model}"  
        styleClass="optional-css-name"/>
```

35

Syntax Summary: Java (Part 1 – LineChartSeries)

```
private void initializeSeries1() {  
    series1 = new LineChartSeries();  
    series1.setLabel("Series 1");  
    series1.set("Q1", 2);  
    series1.set("Q2", 3);  
    ...  
}
```

```
private void initializeSeries1() {  
    series2 = new LineChartSeries();  
    series2.setLabel("Series 1");  
    series2.set("Q1", 5);  
    series1.set("Q2", 3);  
    ...  
}
```

Same in every way as for non-filled line charts

36

Syntax Summary: Java (Part 2 – LineChartModel)

```
private void initializeModel() {
    model = new LineChartModel();
    setAxisInfo();
    series1.setFill(true);
    model.addSeries(series1);
    series2.setFill(true);
    model.addSeries(series2);
    ...
    model.setStacked(true);
}

public LineChartModel getModel() {
    return(model);
}
```

37

Syntax Summary: Java (Part 3 – Axis Information)

```
private void setAxisInfo() {
    model.setTitle("Overall Graph Title");
    model.setLegendPosition("e"); // Or w, s, etc.
    model.setShowPointLabels(true);
    model.getAxes()
        .put(AxisType.X,
            new CategoryAxis("x-axis-title"));
    Axis yAxis = model.getAxis(AxisType.Y);
    yAxis.setMin(minValue);
    yAxis.setMax(maxValue);
    yAxis.setLabel("y-axis-title");
    ...
}
```

Same in every way as for non-filled line charts

38

Example: HTML

```
<p:fieldset legend="Area Chart">
  <p:chart type="line"
           model="#{areaChart.yearModel}"/>
</p:fieldset>
```

39

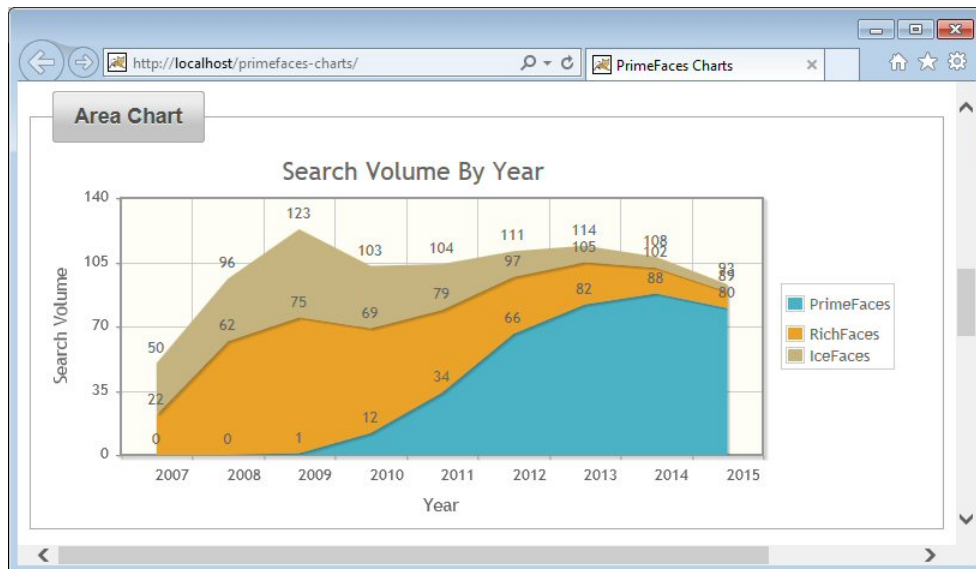
Example: Java (Model)

```
@ManagedBean
@ApplicationScoped
public class AreaChart extends LineChart {
    public AreaChart() {
        LineChartModel yearModel = getYearModel();
        List<ChartSeries> seriesList = yearModel.getSeries();
        for(ChartSeries series: seriesList) {
            LineChartSeries lcSeries = (LineChartSeries)series;
            lcSeries.setFill(true);
        }
        yearModel.getAxis(AxisType.Y).setMax(140);
        yearModel.setStacked(true);
    }
}
```

I just extend the previous line chart example, call `setFill(true)` on each of the `LineChartSeries`, change the maximum y value slightly, and call `setStacked(true)` on the main model.

40

Example: Results



41

© 2015 Marty Hall



Bar Charts



Customized Java EE Training: <http://courses.coreservlets.com/>

Java 7, Java 8, JSF 2, PrimeFaces, Android, JSP, Ajax, jQuery, Spring MVC, RESTful Web Services, GWT, Hadoop
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

Overview

- **Java**
 - Make BarChartSeries
 - Same basic syntax as for LineChartSeries: make BarChartSeries and call set(xValue, yValue) repeatedly
 - Make BarChartModel
 - Set axis information and legend position
 - Mostly same as for line charts, but simpler X axis
 - Call addSeries for each BarChartSeries from above
 - Same basic syntax as for LineChartModel
- **HTML (p:chart)**
 - type: "bar"
 - model: the BarChartModel from above

43

Syntax Summary: HTML

```
<p:chart type="bar"  
        model="#{someBean.model}"  
        styleClass="optional-css-name"/>
```

44

Syntax Summary: Java (Part 1 – BarChartSeries)

```
private void initializeSeries1() {
    series1 = new BarChartSeries();
    series1.setLabel("Series 1");
    series1.set("Q1", 2);
    series1.set("Q2", 3);
    ...
}
```

```
private void initializeSeries1() {
    series2 = new BarChartSeries();
    series2.setLabel("Series 1");
    series2.set("Q1", 5);
    series1.set("Q2", 3);
    ...
}
```

Same as for line charts except the type is BarChartSeries instead of LineChartSeries.

You can also use ChartSeries here instead of BarChartSeries, but you cannot use ChartSeries instead of LineChartSeries for the two previous line examples. So, despite the identical syntax, you cannot completely share the code for the series data for a line chart and a bar chart.

45

Syntax Summary: Java (Part 2 – BarChartModel)

```
private void initializeModel() {
    model = new BarChartModel();
    setAxisInfo();
    model.addSeries(series1);
    model.addSeries(series2);
    ...
}

public BarChartModel getModel() {
    return(model);
}
```

46

Syntax Summary: Java (Part 3 – Axis Information)

```
private void setAxisInfo() {
    model.setTitle("Overall Graph Title");
    model.setLegendPosition("e"); // Or w, s, etc.
    Axis xAxis = yearModel.getAxis(AxisType.X);
    xAxis.setLabel("Year");
    Axis yAxis = model.getAxis(AxisType.Y);
    yAxis.setMin(minValue);
    yAxis.setMax(maxValue);
    yAxis.setLabel("y-axis-title");
    ...
}
```

You don't call `setShowPointLabels`, and you get the X axis and set its label in a slightly different way. Otherwise, this is the same as for line charts.

47

Example: HTML

```
<p:fieldset legend="Bar Chart">
  <p:chart type="bar"
    model="#{barChart.yearModel}" />
</p:fieldset>
```

48

Example: Java (Chart Data)

```
private void buildBarChartSeriesFromLineChartSeries() {
    primeFacesSeries2 = new BarChartSeries();
    primeFacesSeries2.setData(primeFacesSeries.getData());
    primeFacesSeries2.setLabel(primeFacesSeries.getLabel());
    richFacesSeries2 = new BarChartSeries();
    richFacesSeries2.setData(richFacesSeries.getData());
    richFacesSeries2.setLabel(richFacesSeries.getLabel());
    iceFacesSeries2 = new BarChartSeries();
    iceFacesSeries2.setData(iceFacesSeries.getData());
    iceFacesSeries2.setLabel(iceFacesSeries.getLabel());
}
```

You cannot use a `LineChartSeries` in a bar chart. But, it is repetitive to repeat practically identical code. So, I just copy the data and labels from my previous `LineChartSeries`.

If you have a bar chart that does not share the data points with a line chart, then just make a `BarChartSeries`, set the label, and repeatedly call `set(xValue, yValue)`, just as with a `LineChartSeries`.

49

Example: Java (Model)

```
@ManagedBean
@ApplicationScoped
public class BarChart {
    private BarChartModel yearModel;

    public BarChart() {
        yearModel = new BarChartModel();
        setAxisInfo();
        ChartData sampleData = new ChartData();
        BarChartSeries primeFacesSeries = sampleData.primeFacesSeries2();
        yearModel.addSeries(primeFacesSeries);
        BarChartSeries richFacesSeries = sampleData.richFacesSeries2();
        yearModel.addSeries(richFacesSeries);
        BarChartSeries iceFacesSeries = sampleData.iceFacesSeries2();
        yearModel.addSeries(iceFacesSeries);
    }

    public BarChartModel getYearModel() {
        return(yearModel);
    }
}
```

Except for the highlighted class names, practically the same as the first line chart example.

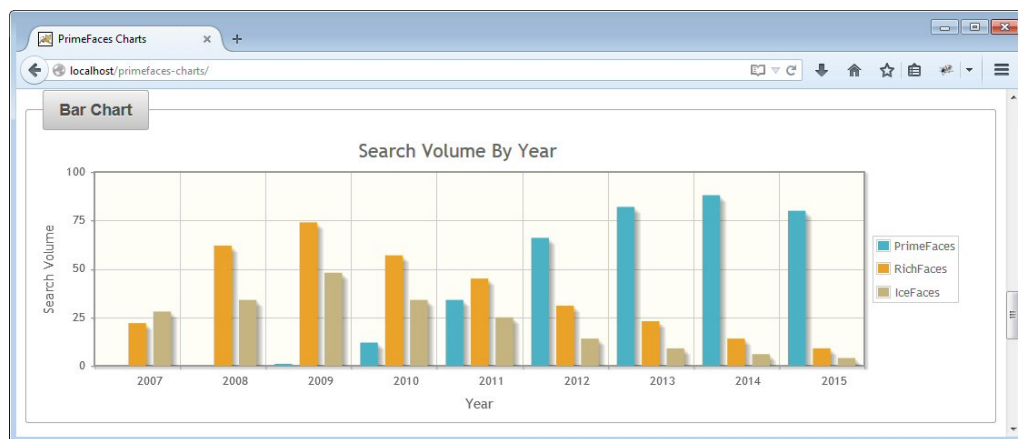
50

Example: Java (Axis Info)

```
private void setAxisInfo() {  
    yearModel.setTitle("Search Volume By Year");  
    yearModel.setLegendPosition("e");  
    Axis xAxis = yearModel.getAxis(AxisType.X);  
    xAxis.setLabel("Year");  
    Axis yAxis = yearModel.getAxis(AxisType.Y);  
    yAxis.setMin(0);  
    yAxis.setMax(100);  
    yAxis.setLabel("Search Volume");  
}
```

51

Example: Results



52



Interactive (Ajax-Enabled) Charts



Customized Java EE Training: <http://courses.coreservlets.com/>

Java 7, Java 8, JSF 2, PrimeFaces, Android, JSP, Ajax, jQuery, Spring MVC, RESTful Web Services, GWT, Hadoop
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

Overview

- **Idea: you can fire Ajax requests on clicks**
 - You designate a server-side method that takes an `ItemSelectEvent` as an argument. From the event, you can get two important values
 - Series index (`event.getSeriesIndex()`)
 - Item index (`event.getItemIndex()`)

- **Java**

```
public void someMethod(ItemSelectEvent event) { ... }
```

- **HTML**

```
<h:form><p:chart ...>
<p:ajax event="itemSelect"
        listener="#{someBean.someMethod}"
        update="..." />
</p:chart></h:form>
```

Notice that the chart must be inside `h:form`.
Non-interactive charts have no such requirement.

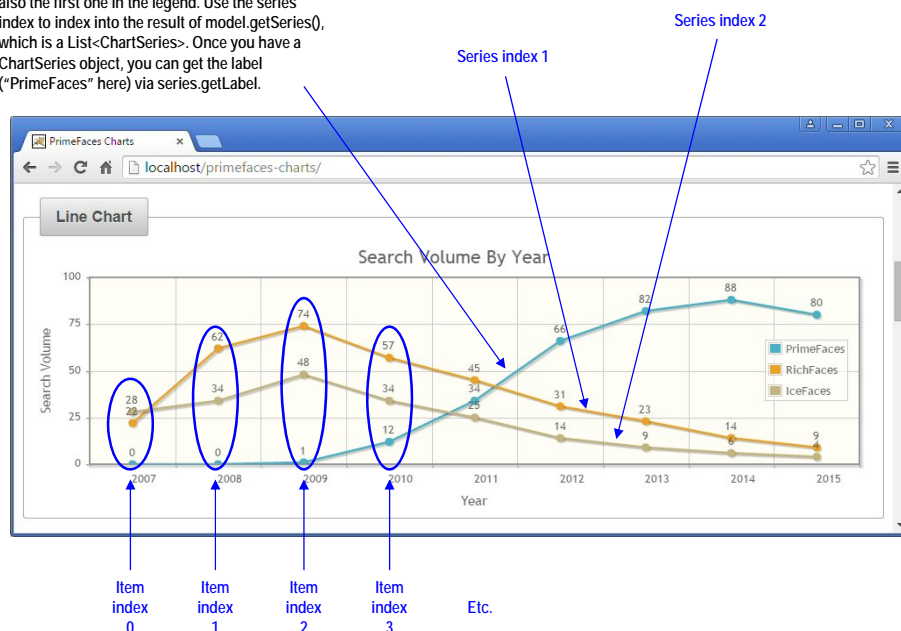
Series Index and Item Index: Overview

- **For line chart**
 - The series index identifies the line. For a bar chart, this identifies which set of bars (color) was clicked.
 - The item index identifies the x value in that line (and you can look up the y value based on that x value)
- **For bar chart**
 - The series index identifies the bar color
 - The item index identifies the x value: the set of bars (and you can look up the y value [bar height] based on that x value)
- **For pie chart**
 - The series index is always 0
 - The item index identifies the value (size of pie wedge)

55

Series Index and Item Index: Line Charts

Series index 0. The PrimeFaces series was the first one added to the model with addSeries and is also the first one in the legend. Use the series index to index into the result of model.getSeries(), which is a List<ChartSeries>. Once you have a ChartSeries object, you can get the label ("PrimeFaces" here) via series.getLabel.



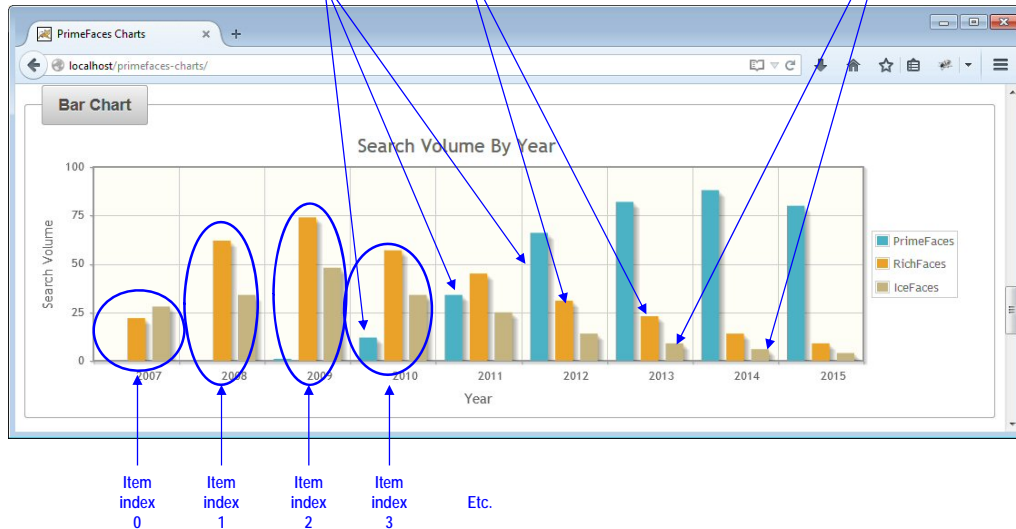
56

Series Index and Item Index: Bar Charts

All blue bars have series index 0. The PrimeFaces series was the first one added to the model with addSeries and is also the first one in the legend. Use the series index to index into the result of model.getSeries(), which is a List<ChartSeries>. Once you have a ChartSeries object, you can get the label ("PrimeFaces" here) via series.getLabel.

All orange bars have series index 1

All tan bars have series index 2



57

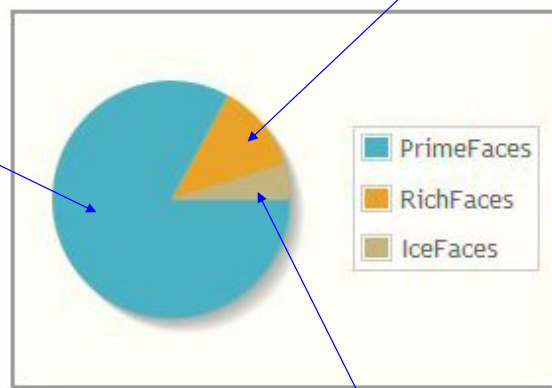
Series Index and Item Index: Pie Charts

The series index is 0 in all cases, since there is only one set of data.

Item index 1. The second call to PieChartModel.set was model2015.set("RichFaces", 10).

Item index 0. The first call to PieChartModel.set was model2015.set("PrimeFaces", 68).

2015 Search Volume



Item index 2. The third call to PieChartModel.set was model2015.set("IceFaces", 4).

58

Line and Bar Charts: Getting Y Value from Series and Item Index

- **Get ChartSeries object from series index**
 - `model.getSeries` returns a `List<ChartSeries>`, use `thatList.get(seriesIndex)` to get `ChartSeries` object
- **Get x value from item index**
 - Transform the item index into the value actually used when you called `yourChartSeries.set(xValue, yValue)`
 - E.g., in our examples, if the index is 0, the x value is 2007, if the index is 1, the x value is 2008, etc. So, use `String.valueOf(2007 + itemIndex)`.
- **Get Map that links x values to y values**
 - `Map<Object,Number> seriesMap = selectedSeries.getData();`
- **Get the y value**
 - `seriesMap.get(xValue)`

59

Example: HTML

```
<h:form>
<p:growl id="growl"/>
<p:chart type="bar" model="#{ajaxBarChart.yearModel}">
  <p:ajax event="itemSelect"
    listener="#{ajaxBarChart.handleItemSelect}"
    update="growl"/>
</p:chart>
</h:form>
```

Notice that the chart must be inside `h:form`.
Non-interactive charts have no such requirement.

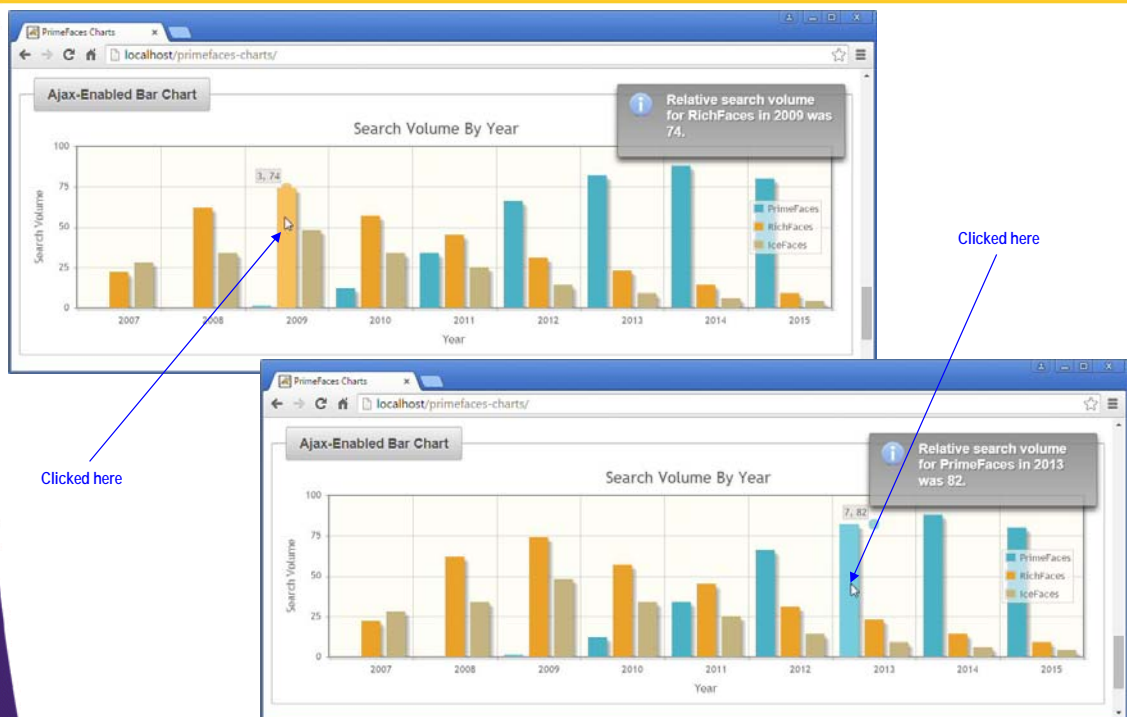
60

Example: Java

```
@ManagedBean
@ApplicationScoped
public class AjaxBarChart extends BarChart {
    public void handleItemSelect(ItemSelectEvent event) {
        BarChartModel model = getYearModel(); // From parent class
        int seriesIndex = event.getSeriesIndex();
        int itemIndex = event.getItemIndex();
        List<ChartSeries> allSeries = model.getSeries();
        ChartSeries selectedSeries = allSeries.get(seriesIndex);
        Map<Object,Number> seriesMap = selectedSeries.getData();
        String componentLibraryName = selectedSeries.getLabel();
        String mapKey = String.valueOf(ChartData.START_YEAR + itemIndex);
        Number mapValue = seriesMap.get(mapKey);
        String messageText =
            String.format("Relative search volume for %s in %s was %s.",
                componentLibraryName, mapKey, mapValue);
        FacesMessage message = new FacesMessage(messageText);
        message.setSeverity(FacesMessage.SEVERITY_INFO);
        FacesContext.getCurrentInstance().addMessage(null, message);
    }
}
```

61

Example: Results



62



Wrap-Up



Customized Java EE Training: <http://courses.coreservlets.com/>

Java 7, Java 8, JSF 2, PrimeFaces, Android, JSP, Ajax, jQuery, Spring MVC, RESTful Web Services, GWT, Hadoop
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

Summary: Line Charts

- **Make series**

```
LineChartSeries data1 = new LineChartSeries();  
data1.add(xValue1, yValue1);  
data1.add(xValue2, yValue2);  
...  
LineChartSeries data2 = new LineChartSeries();  
...
```

- **Make model**

```
LineChartModel model = LineChartModel();  
model.addSeries(data1);  
model.addSeries(data2);  
setUpAxisInformation();
```

- **Refer to model in HTML**

```
<p:chart type="line" model="#{chartBean.model}"/>
```

Summary: Other Charts

- **Area charts**
 - Call `setFilled(true)` on each of the series
 - Call `setStacked(true)` on the model
- **Bar charts**
 - On previous slide, replace `LineChartSeries` with `BarChartSeries` and `LineChartModel` with `BarChartModel`
- **Interactive charts**
 - Use `p:ajax` to refer to method that takes `ItemSelectEvent`
 - Use series index and item index to get y value
- **Pie charts**
 - No separate series. Call `set(wedgeLabel, wedgeValue)`

65

© 2015 Marty Hall



Questions?

More info:

<http://www.coreservlets.com/JSF-Tutorial/jsf2/> – JSF 2.2 tutorial

<http://www.coreservlets.com/JSF-Tutorial/primefaces/> – PrimeFaces tutorial

<http://courses.coreservlets.com/jsf-training.html> – Customized JSF and PrimeFaces training courses

<http://coreservlets.com/> – JSF 2, PrimeFaces, Java 7 or 8, Ajax, jQuery, Hadoop, RESTful Web Services, Android, HTML5, Spring, Hibernate, Servlets, JSP, GWT, and other Java EE training



Customized Java EE Training: <http://courses.coreservlets.com/>

Java 7, Java 8, JSF 2, PrimeFaces, Android, JSP, Ajax, jQuery, Spring MVC, RESTful Web Services, GWT, Hadoop

Developed and taught by well-known author and developer. At public venues or onsite at *your* location.