```
/*

    Document   : Demo1
    Created on : Mar 22, 2012, 11:20:14 AM
    Description: Style our JavaFX 2 application demo.
                 Put this css style sheet in same package
                 (or dictory) as Java class.
*/


/**
 * This is the root node. In demo this is the BorderPane that is passed
 * into the Scene constructor.
 */
#appContainer {
    -fx-background-color: linear-gradient(to bottom, #464646, #5f5f5f);
}


/**
 * This CSS style is for the buttonBox in Left region of BorderPane.
 * It sets a linear gradient background color and applies a 1px padding
 * around the VBox inner edges.
 */
#buttonMenuContainer {
    -fx-background-color: linear-gradient(to bottom, #737373, #595959);
    -fx-padding: 10px;
}


/**
 * Give the buttonBox container a subtle hover effect.
 */
#buttonMenuContainer:hover {
    -fx-background-color: linear-gradient(to bottom, #717171, #515151);
}


/**
 * This is a makeshift border. #divider1 creates the right border inside
 * The Application's Left Region. #divider2 creates the bottom border
 * for the title.
 *
 * The idea was to use a Blank Label. For example, for the Title, which is
 * located in the BorderPane Top Region; I've set the Labels minWidth
 * property to the screen size. This creates the effect of a bottom border.
 *
 * Set the background color here.
```

```css
 */
#divider1, #divider2{
    -fx-background-color: linear-gradient(to bottom, #dad9d9, #ffffff);
}


/**
 * the topLayoutContainer is a BorderPane manager that is placed inside the
 * application's layout container (root node). This is how I was able to
 * create the bottom border effect (#divider2).
 */
#topLayoutContainer {
    -fx-background-color: linear-gradient(to bottom, #ebeced, #b5bcc6);
}


/**
 * appTitle text.
 */
#appTitle {
    -fx-text-fill: linear-gradient(to bottom, #717171, #515151);
    -fx-font-family: verdana;
    -fx-font-size: 24px;
}


/**
 * This class is a HBox used to set the alignment of the appTitle Text
 * The padding is set to push the title down and to the right from the
 * container edges.
 */
#titleBox {
    -fx-padding: 10px 0 0 15px;
}


/*
 * In this css style, we are applying styles to a GridPane.
 * We apply a border with radius. We also need to apply the
 * background radius. This creates the rounded corners when
 * the first button is clicked. Remember to set both background
 * and border radius.
 */
#gridPane_Example1 {
    -fx-border-color: #ffffff;
    -fx-border-width: 1px;
    -fx-border-radius: 4;
```

```
    -fx-background-radius: 4;
    -fx-background-color: linear-gradient(to bottom, #b5bcc6, #dee3e4);
    -fx-padding: 15;


}


/**
 * exContainer ID is applied to a VBox. This VBox is what is physically
 * loaded into the Applications Center region (root node). I use the VBox
 * so content automatically fills available space. 100% width and height.
 */
#exContainer {
    -fx-padding: 10px;
}


/**
 * First Name TextField.
 */
#txtFirstName {
    -fx-border-color: #939393;
    -fx-border-radius: 5px;
    -fx-background-radius: 5px;
    -fx-background-color: linear-gradient(to bottom, #cccccc, #f2f2f2);
}


/**
 * The focused effect.
 */
#txtFirstName:focused {
    -fx-border-color: linear-gradient(to left, #ff0000, #7d1e15);
}


/**
 * The hover effect.
 */
#txtFirstName:hover {
    -fx-border-color: linear-gradient(to left, #ffca45, #f8a70f);
}


/**
 * Keep the focused coloring when mouse is still hovering over
 * text field while active.
 */
```

```
#txtFirstName:focused:hover {
    -fx-border-color: linear-gradient(to left, #ff0000, #7d1e15);
}


/**
 * Apply this style to all Label objects.
 */
Label {
    -fx-font-size: 12px;
    -fx-font-weight: lighter;
}
```