Expression Language, EL II

Accessing Collections

Equivalence of Dot and Array Notations

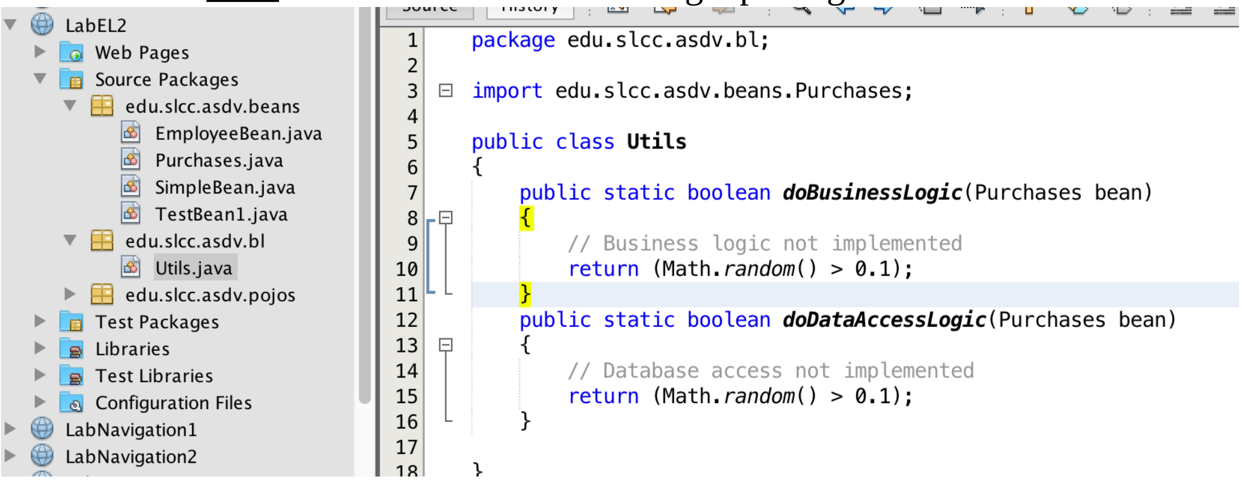
- Equivalent forms
  - #{name.property}
    - Legal ONLY if “property” is **LEGAL** Java variable name
  - #{name["property"]}

- Reasons for using bracket notation
  - To access arrays, lists, and other collections

  - To calculate the property name at request time.
    - #{name1[name2]} (no quotes around name2)

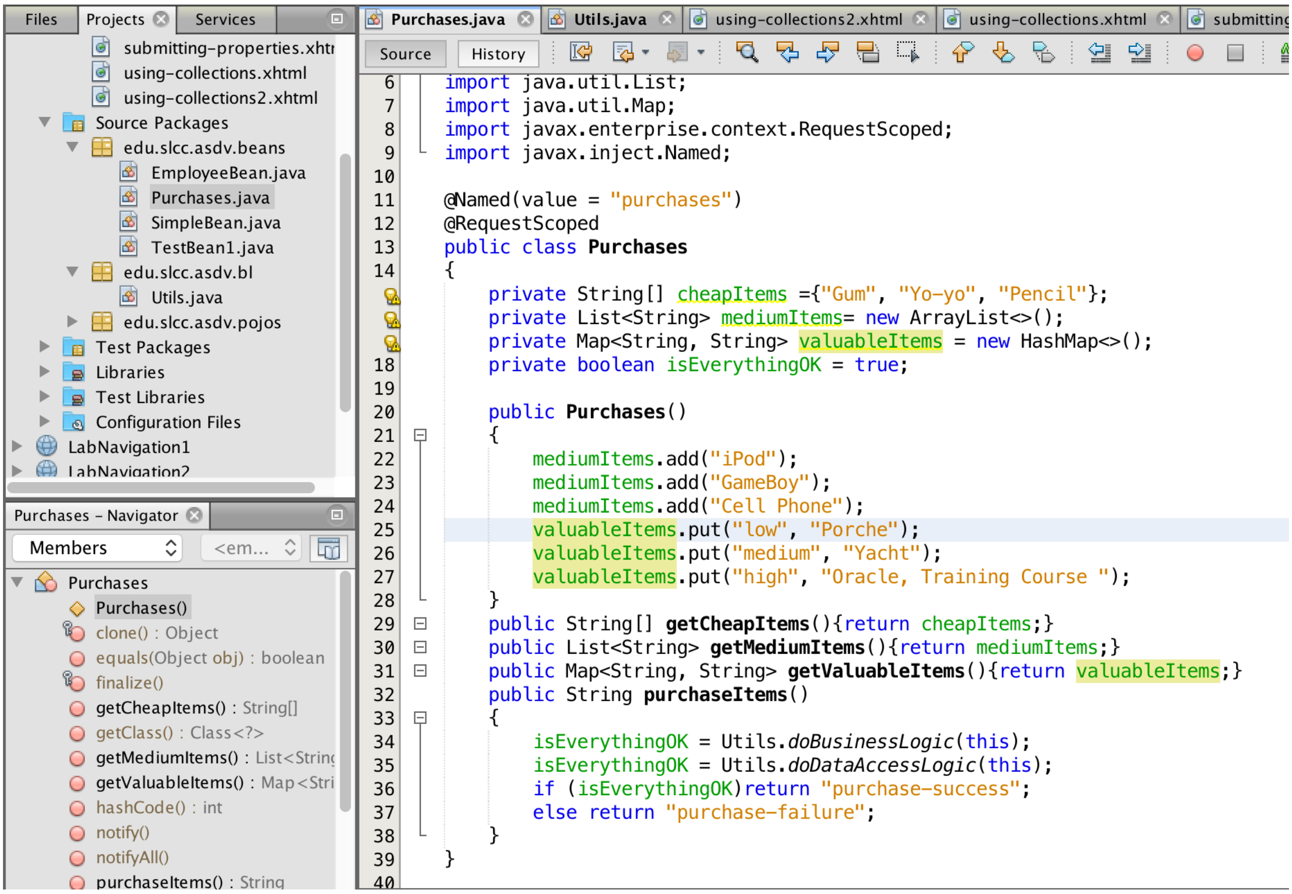
  - To use names that are **ILLEGAL** as Java variable names
    - #{foo["bar-baz"]}
    - #{foo["bar.baz"]}

1. Copy LabeEL1 and rename it LabEL2
2. Create a Utils class under the business logic package



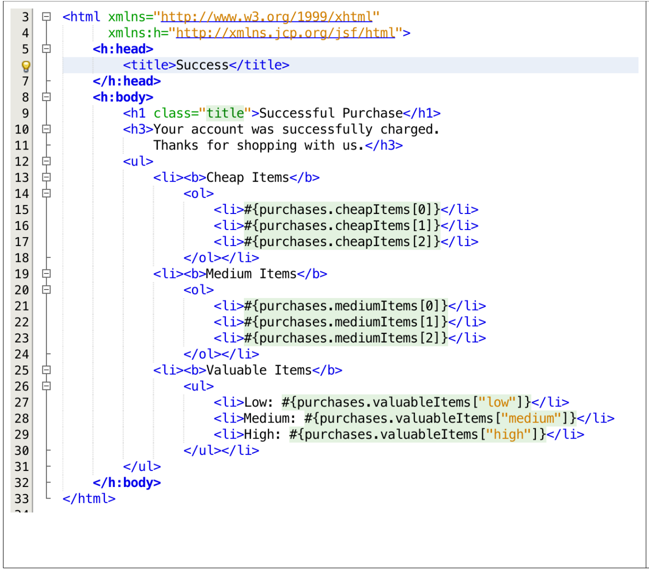
```java
package edu.slcc.asdv.bl;

import edu.slcc.asdv.beans.Purchases;

public class Utils
{
    public static boolean doBusinessLogic(Purchases bean)
    {
        // Business logic not implemented
        return (Math.random() > 0.1);
    }
    public static boolean doDataAccessLogic(Purchases bean)
    {
        // Database access not implemented
        return (Math.random() > 0.1);
    }

}
```

3. Create a request scoped bean Purchases that has an Array, a List and a Map.

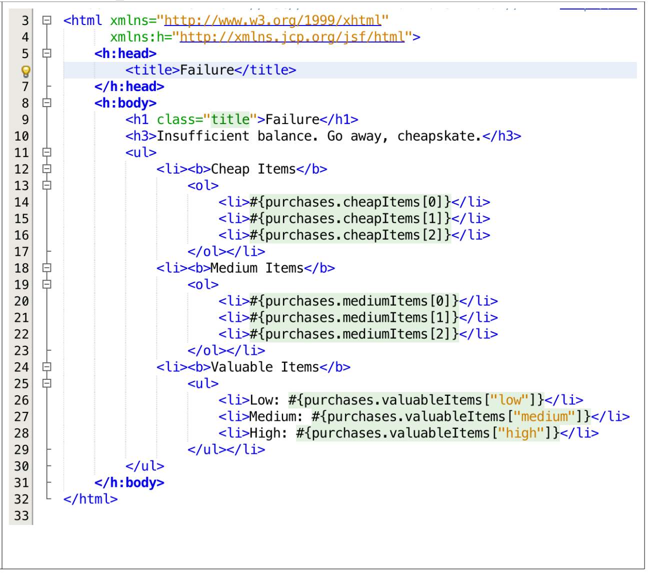


```java
import java.util.List;
import java.util.Map;
import javax.enterprise.context.RequestScoped;
import javax.inject.Named;

@Named(value = "purchases")
@RequestScoped
public class Purchases
{
    private String[] cheapItems ={"Gum", "Yo-yo", "Pencil"};
    private List<String> mediumItems= new ArrayList<>();
    private Map<String, String> valuableItems = new HashMap<>();
    private boolean isEverythingOK = true;

    public Purchases()
    {
        mediumItems.add("iPod");
        mediumItems.add("GameBoy");
        mediumItems.add("Cell Phone");
        valuableItems.put("low", "Porche");
        valuableItems.put("medium", "Yacht");
        valuableItems.put("high", "Oracle, Training Course ");
    }
    public String[] getCheapItems(){return cheapItems;}
    public List<String> getMediumItems(){return mediumItems;}
    public Map<String, String> getValuableItems(){return valuableItems;}
    public String purchaseItems()
    {
        isEverythingOK = Utils.doBusinessLogic(this);
        isEverythingOK = Utils.doDataAccessLogic(this);
        if (isEverythingOK)return "purchase-success";
        else return "purchase-failure";
    }
}
```

4. Add JSF pages purchase-success.xhtml and purchase-failure.xhtml.

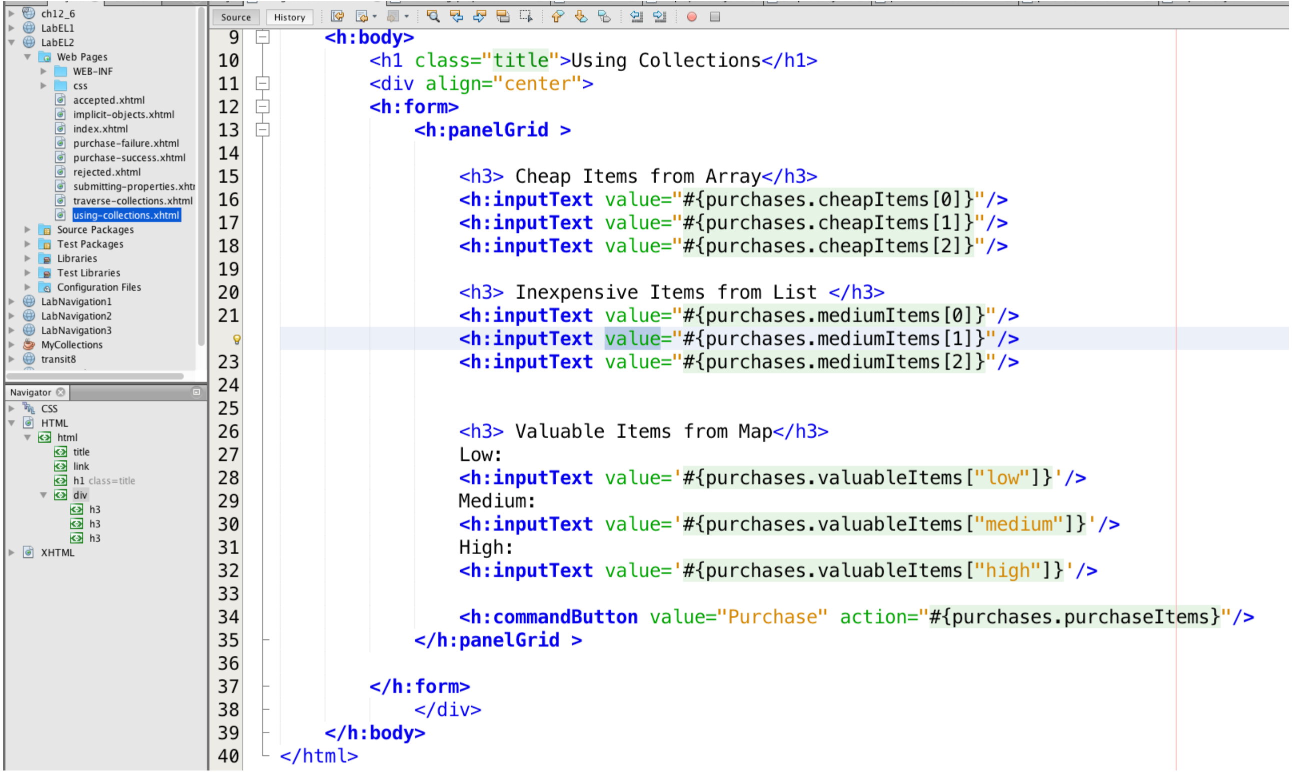


```xml
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://xmlns.jcp.org/jsf/html">
    <h:head>
        <title>Success</title>
    </h:head>
    <h:body>
        <h1 class="title">Successful Purchase</h1>
        <h3>Your account was successfully charged.
            Thanks for shopping with us.</h3>
        <ul>
            <li><b>Cheap Items</b>
                <ol>
                    <li>#{purchases.cheapItems[0]}</li>
                    <li>#{purchases.cheapItems[1]}</li>
                    <li>#{purchases.cheapItems[2]}</li>
                </ol></li>
            <li><b>Medium Items</b>
                <ol>
                    <li>#{purchases.mediumItems[0]}</li>
                    <li>#{purchases.mediumItems[1]}</li>
                    <li>#{purchases.mediumItems[2]}</li>
                </ol></li>
            <li><b>Valuable Items</b>
                <ul>
                    <li>Low: #{purchases.valuableItems["low"]}</li>
                    <li>Medium: #{purchases.valuableItems["medium"]}</li>
                    <li>High: #{purchases.valuableItems["high"]}</li>
                </ul></li>
        </ul>
    </h:body>
</html>
```



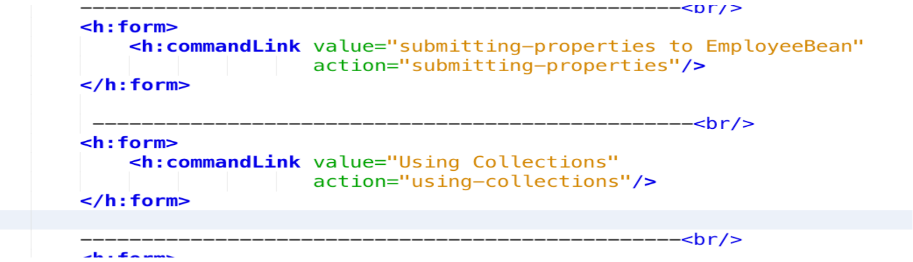
```xml
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://xmlns.jcp.org/jsf/html">
    <h:head>
        <title>Failure</title>
    </h:head>
    <h:body>
        <h1 class="title">Failure</h1>
        <h3>Insufficient balance. Go away, cheapskate.</h3>
        <ul>
            <li><b>Cheap Items</b>
                <ol>
                    <li>#{purchases.cheapItems[0]}</li>
                    <li>#{purchases.cheapItems[1]}</li>
                    <li>#{purchases.cheapItems[2]}</li>
                </ol></li>
            <li><b>Medium Items</b>
                <ol>
                    <li>#{purchases.mediumItems[0]}</li>
                    <li>#{purchases.mediumItems[1]}</li>
                    <li>#{purchases.mediumItems[2]}</li>
                </ol></li>
            <li><b>Valuable Items</b>
                <ul>
                    <li>Low: #{purchases.valuableItems["low"]}</li>
                    <li>Medium: #{purchases.valuableItems["medium"]}</li>
                    <li>High: #{purchases.valuableItems["high"]}</li>
                </ul></li>
        </ul>
    </h:body>
</html>
```

5. Create page using-collections.



```
    <h:body>
        <h1 class="title">Using Collections</h1>
        <div align="center">
        <h:form>
            <h:panelGrid >

                <h3> Cheap Items from Array</h3>
                <h:inputText value="#{purchases.cheapItems[0]}"/>
                <h:inputText value="#{purchases.cheapItems[1]}"/>
                <h:inputText value="#{purchases.cheapItems[2]}"/>

                <h3> Inexpensive Items from List </h3>
                <h:inputText value="#{purchases.mediumItems[0]}"/>
                <h:inputText value="#{purchases.mediumItems[1]}"/>
                <h:inputText value="#{purchases.mediumItems[2]}"/>


                <h3> Valuable Items from Map</h3>
                Low:
                <h:inputText value='#{purchases.valuableItems["low"]}'/>
                Medium:
                <h:inputText value='#{purchases.valuableItems["medium"]}'/>
                High:
                <h:inputText value='#{purchases.valuableItems["high"]}'/>

                <h:commandButton value="Purchase" action="#{purchases.purchaseItems}"/>
            </h:panelGrid >

        </h:form>
            </div>
    </h:body>
</html>
```

6. Call it from index



```
-------------------------------------------------<br/>
<h:form>
    <h:commandLink value="submitting-properties to EmployeeBean"
                   action="submitting-properties"/>
</h:form>

 -------------------------------------------------<br/>
<h:form>
    <h:commandLink value="Using Collections"
                   action="using-collections"/>
</h:form>

-------------------------------------------------<br/>
<h:form>
```