1. **Agenda**:
   1. Use cases and precautions to understand for web bean scopes.
2. **@Requestscope**
   1. A lot of requests in web app and request time is very short so garbage collection will consume time so performance issue.
   2. Don’t put so much logic in the initialization of request bean scoped as for each request, a new bean is created.
   3. **Use cases**:
      1. When you want to store some sensitive information such as login information (username, password).  
         Credentials passed when making login request must be invalidated as soon as request is done.  
         Now, as we’re using request scoped beans for only sensitive info, no limited number of beans and so no performance issue.
3. **@Session**:
   1. Longer life and very less frequently they are created so less garbage collection so better performance.
   2. **Use Cases**:
      1. After successfully login, we want to display username, welcome msg on each subsequent pages.
   3. **Precautions**:
      1. Use store big data, binary data (images) otherwise performance degrade.
4. **@Application**
   1. Very similar to singleton scope.
   2. **Use Cases**:
      1. Drop Down values (Countries : This kind of data doesn’t change as per a user) but be careful and try to fetch from cache or DB.
   3. **Precaution**:
      1. Since singleton in nature, so make sure no race condition.
5. As we know one bean will be created for Singleton scope and Application Scope then what is the difference?  
   Some points to remember:  
   1. One ServletContext per JEE App by definition.

|  |  |
| --- | --- |
| **Singleton Scoped Bean** | **Application Scoped Bean** |
| One bean per Application Context. We know for each configured Dispatcher Servlet, a new application context is created and we can define more than one Dispatcher Servlet in web.xml file. | One bean per ServletContext. |