General

1. URL
   1. Not .do – more realistic (employee/new)
   2. No access to manager pages unless they’re signed in as one
2. Data validation for all pages
   1. Checks if employee or manager
   2. Employee cannot open manager pages
   3. Pages differ depending on login (built on same page)
3. Navbar
   1. In a single file, being imported in all others – reduce code redundancy, easier to edit
   2. Active on current page being used
4. Printing out names
   1. First letter capital, rest lower case \*\*\*\*

JSP Pages:

1. Login.jsp
   1. If logged in, and they press back or if they try going to /login, they will be redirected to index
2. Myinfo.jsp
   1. Uses same edit button as “employee.jsp”
   2. Upon editing someone, page will redirect back to myinfo.jsp, checking if the username is the same as sessionAttribute
3. Employee
   1. Shows them sorted (ORDER BY) USER\_ID
4. Employee/Modify
   1. Sending <input type=”hidden” with user\_id> to update it
   2. Upon clicked “edit”, it checks if the user is a manager, or an employee. It also checks if they’re currently active or inactive. It then changes both the radio and the switch automatically.
   3. Marking an employee “inactive” will “fire” them, but will still have their record
   4. If logged in as a user, they cannot make themselves a manager or inactive
5. Employee/Updated
   1. Uses <c:if tests> to print out new information on that customer (whether they’re a manager or not, active or not, etc.

CSS:

1. Hidden url within web-inf:
   1. href="${pageContext.request.contextPath}/CSS/default.css"

SQL:

1. ERS\_USERS\_ROLES
   1. 3 for “inactive”