Employee data Management Ext JS

Grid Features

Group- 6
March 8 – 2025

Introduction:

- This Document provides a detailed overview of the **Employee data Management** and its significance.
- This System helps Company to add the details of the employee.
- The operations done on the employee are:
  - Create
  - Update
  - Delete
  - Read
- This Insight allow Company to collect the information of employees in tabular data .

- ▶ The **Ext JS Grid** is one of the most powerful and feature-rich components in the Ext JS framework. It is designed to display and manage tabular data efficiently, making it ideal for building data-driven web applications.
- ▶ This is a simple component to display data, which is a collection of record stored in Ext.data.Store in a tabular format.
- ▶ The Ext JS Grid is a versatile and powerful component that provides a rich set of features for displaying and

managing tabular data. Its flexibility, performance, and ease of use make it an excellent choice for building data-driven web applications.

Ext JS Grid Features

- Column Resizing & Reordering
- Data Display: Structured tabular format
- Sorting: Ascending/descending order
- Filtering: Refine data visibility
- Row Selection: Single/multi-row operations
- Editing: Inline editing support
- Customizable UI: Modify styles and themes

Features of the Project

- **Employee List (Grid Panel)** : Show employee details (ID, Name, Department, Email, Phone) in a tabular format.
- **Add Employee :** A modal window with fields for entering employee details (Name, Department, Email, Phone).
- **Edit Employee:** A modal window pre-filled with the selected employee's details.
- **Delete Employee :** A text field to search for employees by name. Remove the employee with a confirmation dialogue.

Layout Structure of the project:

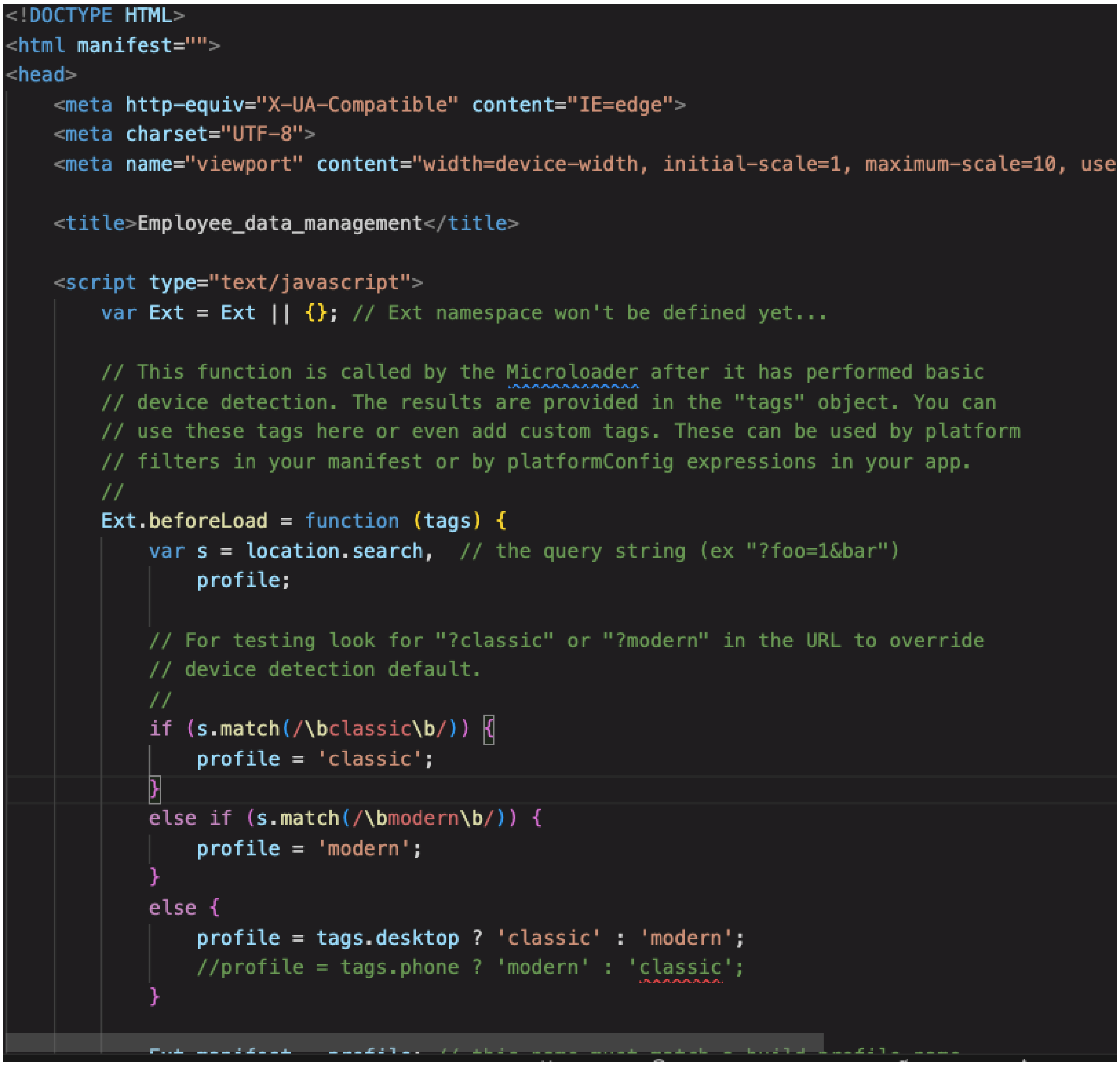
```
/my-ext-project
├── /app
│   ├── /model
│   ├── /store
│   ├── /view
│   ├── /controller
│   └── app.js
├── /ext
│   ├── /build
│   ├── /examples
│   ├── /locale
│   ├── /packages
│   ├── /plugins
│   ├── /resources
│   ├── /src
│   ├── /welcome
│   ├── ext-all.js
│   ├── ext-all-debug.js
│   └── ...
├── /resources
│   └── /css
│       └── app.css
├── /data
│   └── data.json
├── index.html
└── app.json
```

Implementation Steps:

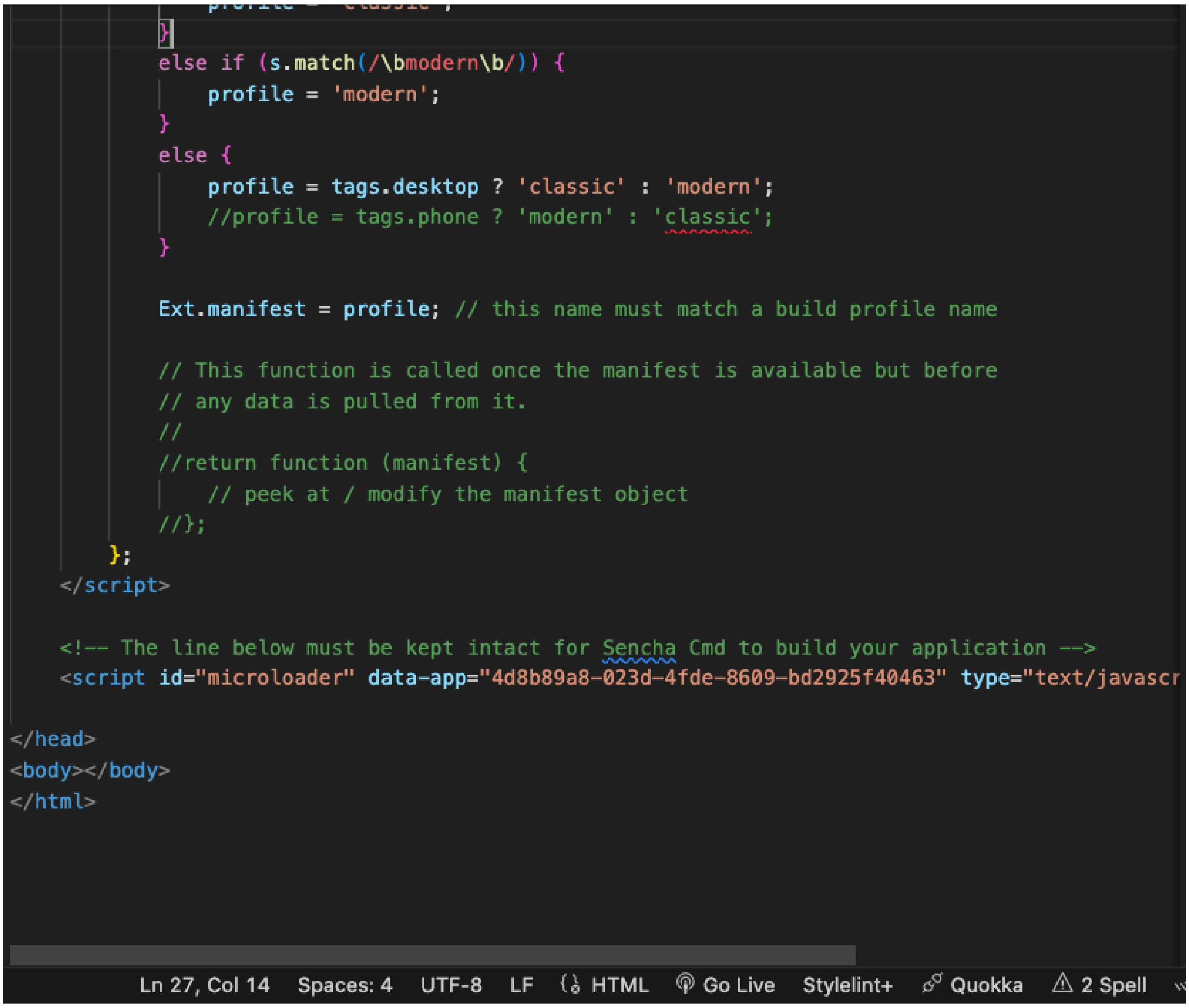
Step1: Setup the Project Structure:

- Create a Project Folder(Employee data management).
- Create Index.html inside the folder.
- Create app.js inside the folder.

Code:



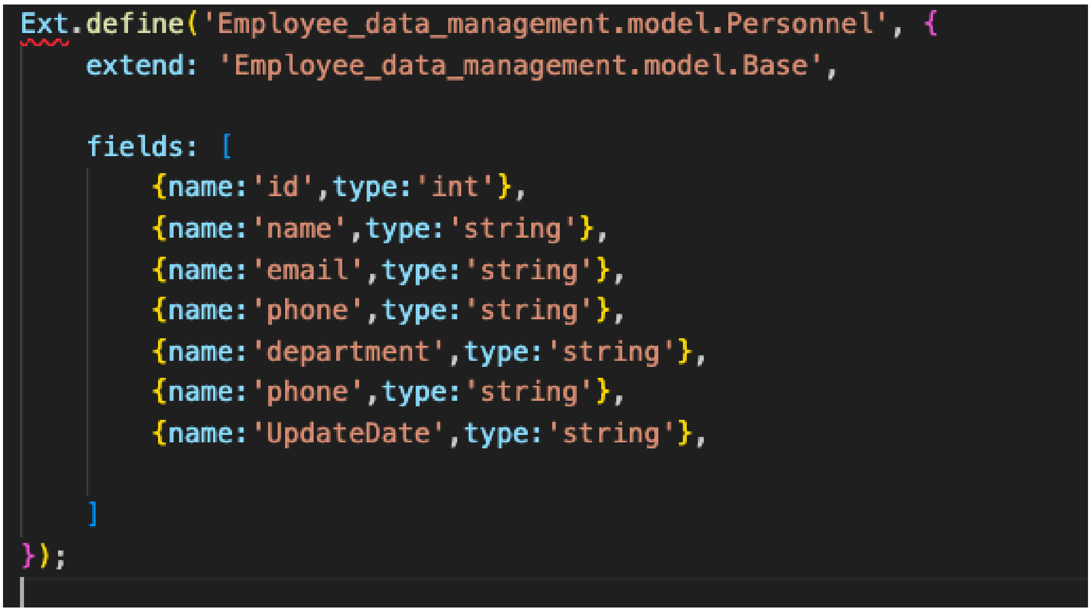
```
<!DOCTYPE HTML>
<html manifest="">
<head>
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=10, use

    <title>Employee_data_management</title>

    <script type="text/javascript">
        var Ext = Ext || {}; // Ext namespace won't be defined yet...

        // This function is called by the Microloader after it has performed basic
        // device detection. The results are provided in the "tags" object. You can
        // use these tags here or even add custom tags. These can be used by platform
        // filters in your manifest or by platformConfig expressions in your app.
        //
        Ext.beforeLoad = function (tags) {
            var s = location.search,  // the query string (ex "?foo=1&bar")
                profile;

            // For testing look for "?classic" or "?modern" in the URL to override
            // device detection default.
            //
            if (s.match(/\bclassic\b/)) {
                profile = 'classic';
            }
            else if (s.match(/\bmodern\b/)) {
                profile = 'modern';
            }
            else {
                profile = tags.desktop ? 'classic' : 'modern';
                //profile = tags.phone ? 'modern' : 'classic';
            }

            Ext.manifest = profile; // this name must match a build profile name
```



```
                profile = 'classic';
            }
            else if (s.match(/\bmodern\b/)) {
                profile = 'modern';
            }
            else {
                profile = tags.desktop ? 'classic' : 'modern';
                //profile = tags.phone ? 'modern' : 'classic';
            }

            Ext.manifest = profile; // this name must match a build profile name

            // This function is called once the manifest is available but before
            // any data is pulled from it.
            //
            //return function (manifest) {
                // peek at / modify the manifest object
            //};
        };
    </script>

    <!-- The line below must be kept intact for Sencha Cmd to build your application -->
    <script id="microloader" data-app="4d8b89a8-023d-4fde-8609-bd2925f40463" type="text/javascr

</head>
<body></body>
</html>
```

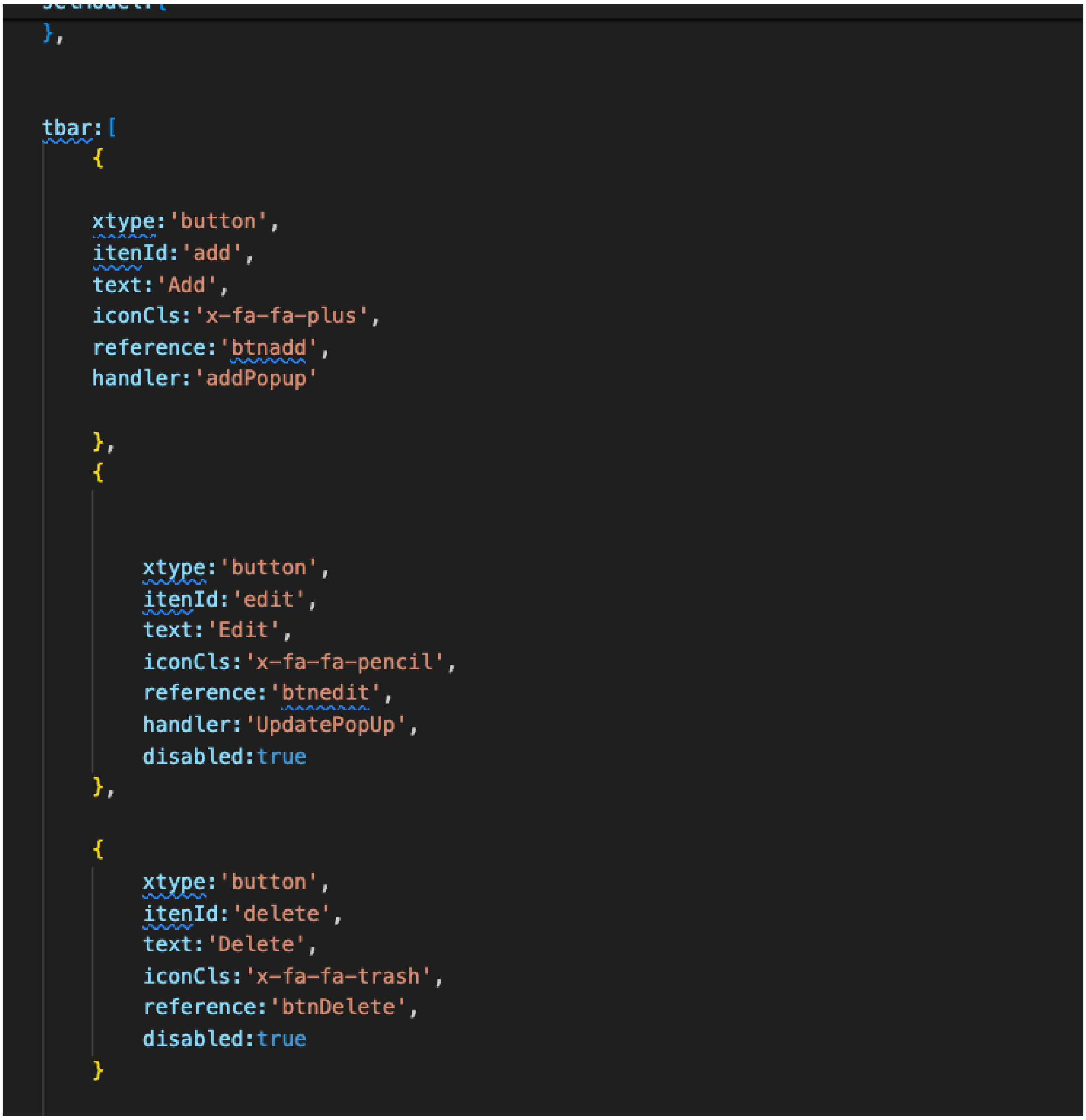
Step 2:

- Adding data into personal.js



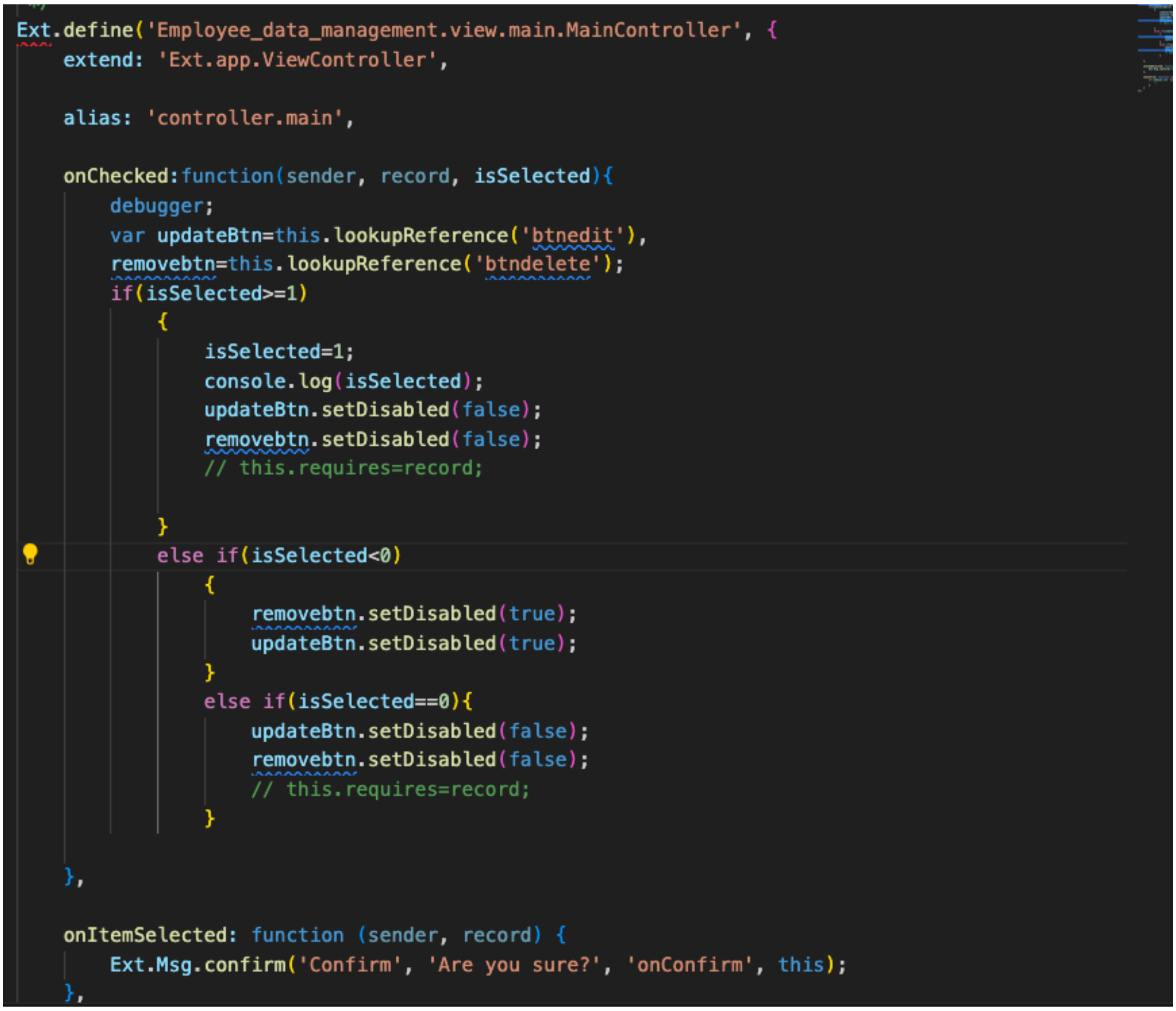
```
Ext.define('Employee_data_management.model.Personnel', {
    extend: 'Employee_data_management.model.Base',

    fields: [
        {name:'id',type:'int'},
        {name:'name',type:'string'},
        {name:'email',type:'string'},
        {name:'phone',type:'string'},
        {name:'department',type:'string'},
        {name:'phone',type:'string'},
        {name:'UpdateDate',type:'string'},

    ]
});
```

- Add Update and delete button in list.js



```
selModel:{
},


tbar:[
    {

    xtype:'button',
    itenId:'add',
    text:'Add',
    iconCls:'x-fa-fa-plus',
    reference:'btnadd',
    handler:'addPopup'

    },
    {


        xtype:'button',
        itenId:'edit',
        text:'Edit',
        iconCls:'x-fa-fa-pencil',
        reference:'btnedit',
        handler:'UpdatePopUp',
        disabled:true
    },

    {
        xtype:'button',
        itenId:'delete',
        text:'Delete',
        iconCls:'x-fa-fa-trash',
        reference:'btnDelete',
        disabled:true
    }
```

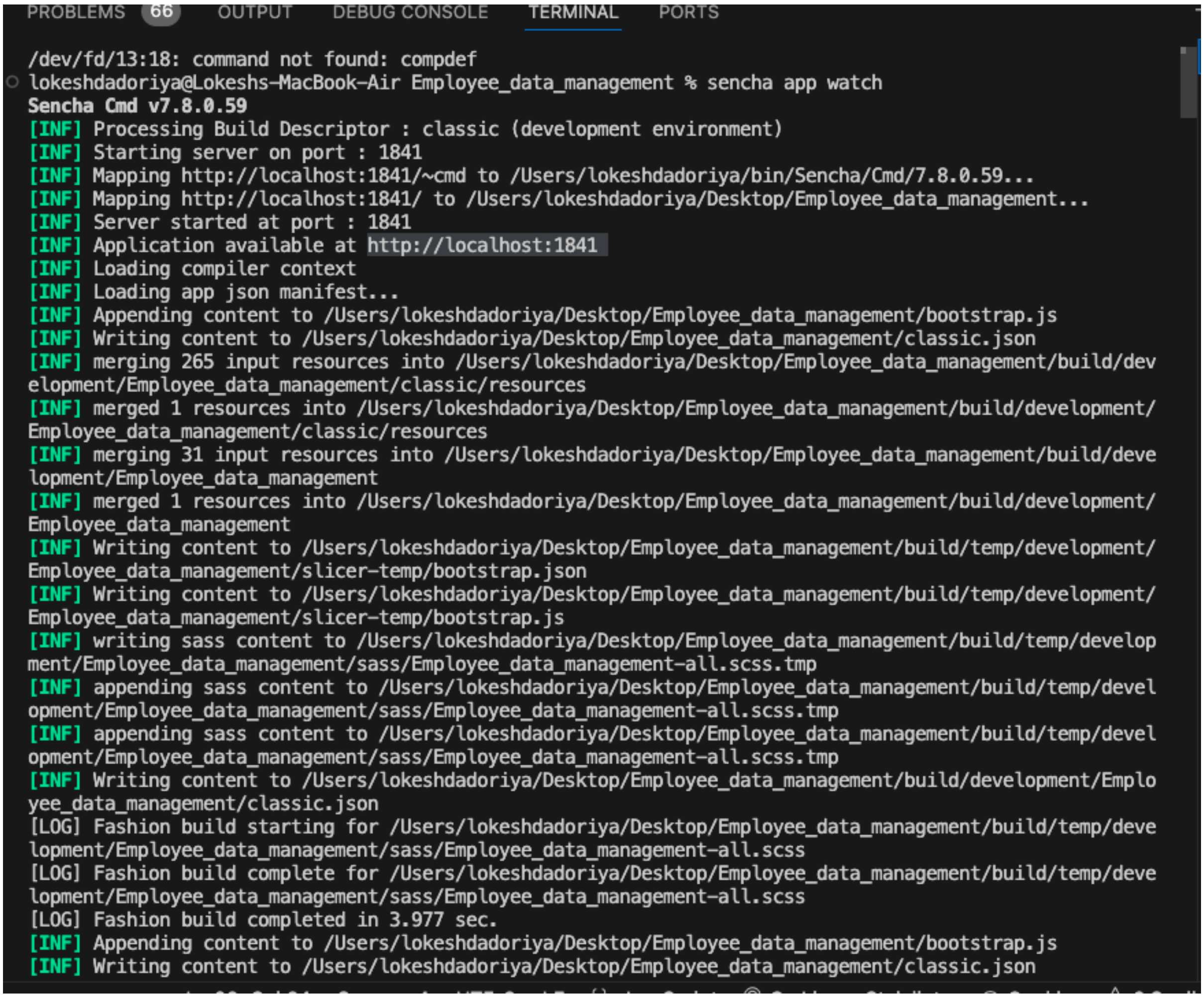
- Implement the logic in the Controller.



```
Ext.define('Employee_data_management.view.main.MainController', {
    extend: 'Ext.app.ViewController',

    alias: 'controller.main',

    onChecked:function(sender, record, isSelected){
        debugger;
        var updateBtn=this.lookupReference('btnedit'),
        removebtn=this.lookupReference('btndelete');
        if(isSelected>=1)
            {
                isSelected=1;
                console.log(isSelected);
                updateBtn.setDisabled(false);
                removebtn.setDisabled(false);
                // this.requires=record;

            }
            else if(isSelected<0)
                {
                    removebtn.setDisabled(true);
                    updateBtn.setDisabled(true);
                }
                else if(isSelected==0){
                    updateBtn.setDisabled(false);
                    removebtn.setDisabled(false);
                    // this.requires=record;
                }

    },

    onItemSelected: function (sender, record) {
        Ext.Msg.confirm('Confirm', 'Are you sure?', 'onConfirm', this);
    },
```

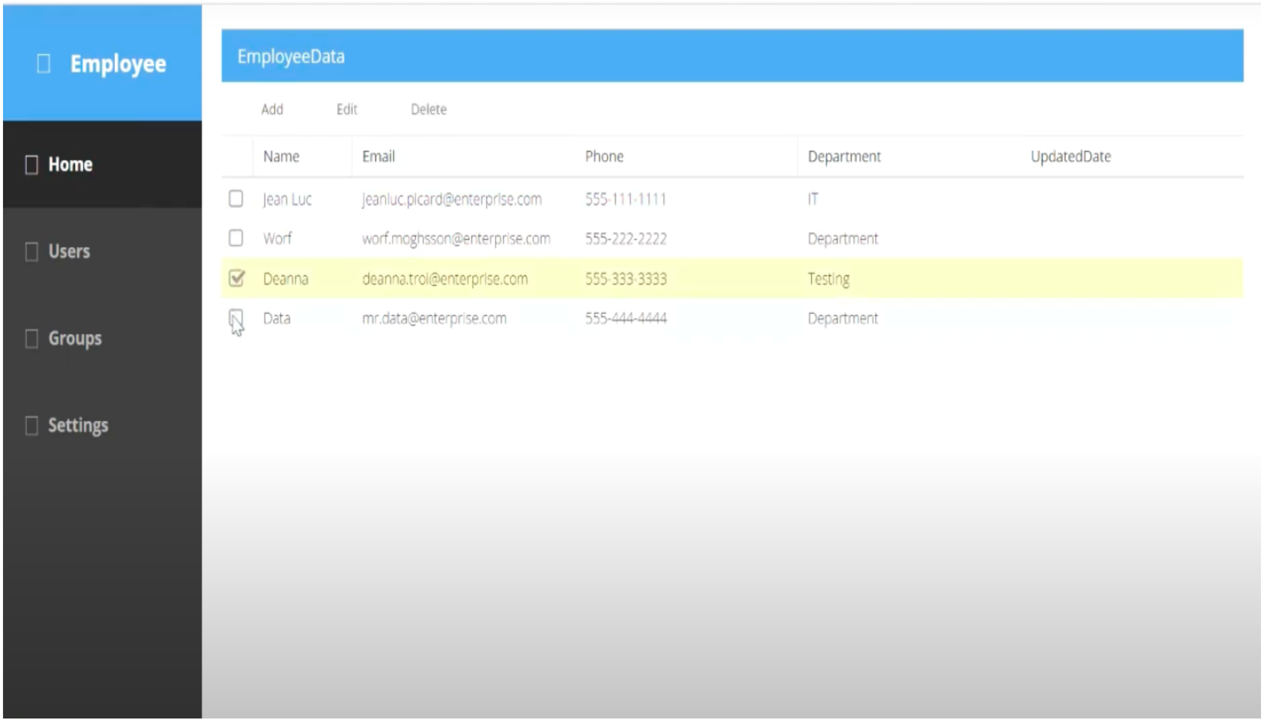
Step 3:

- Run the project by using the command **sencha app watch .**

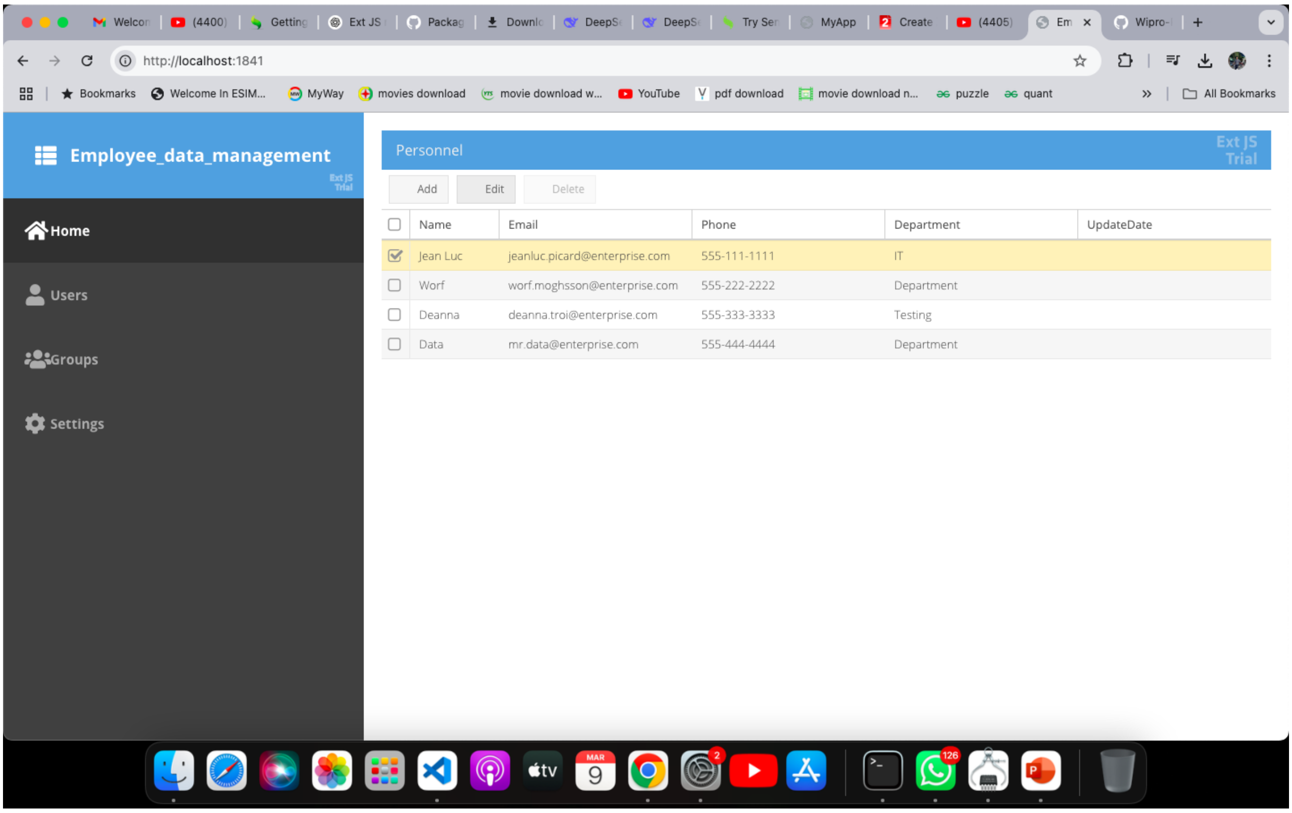


```
/dev/fd/13:18: command not found: compdef
○ lokeshdadoriya@Lokeshs-MacBook-Air Employee_data_management % sencha app watch
Sencha Cmd v7.8.0.59
[INF] Processing Build Descriptor : classic (development environment)
[INF] Starting server on port : 1841
[INF] Mapping http://localhost:1841/~cmd to /Users/lokeshdadoriya/bin/Sencha/Cmd/7.8.0.59...
[INF] Mapping http://localhost:1841/ to /Users/lokeshdadoriya/Desktop/Employee_data_management...
[INF] Server started at port : 1841
[INF] Application available at http://localhost:1841
[INF] Loading compiler context
[INF] Loading app json manifest...
[INF] Appending content to /Users/lokeshdadoriya/Desktop/Employee_data_management/bootstrap.js
[INF] Writing content to /Users/lokeshdadoriya/Desktop/Employee_data_management/classic.json
[INF] merging 265 input resources into /Users/lokeshdadoriya/Desktop/Employee_data_management/build/dev
elopment/Employee_data_management/classic/resources
[INF] merged 1 resources into /Users/lokeshdadoriya/Desktop/Employee_data_management/build/development/
Employee_data_management/classic/resources
[INF] merging 31 input resources into /Users/lokeshdadoriya/Desktop/Employee_data_management/build/deve
lopment/Employee_data_management
[INF] merged 1 resources into /Users/lokeshdadoriya/Desktop/Employee_data_management/build/development/
Employee_data_management
[INF] Writing content to /Users/lokeshdadoriya/Desktop/Employee_data_management/build/temp/development/
Employee_data_management/slicer-temp/bootstrap.json
[INF] Writing content to /Users/lokeshdadoriya/Desktop/Employee_data_management/build/temp/development/
Employee_data_management/slicer-temp/bootstrap.js
[INF] writing sass content to /Users/lokeshdadoriya/Desktop/Employee_data_management/build/temp/develop
ment/Employee_data_management/sass/Employee_data_management-all.scss.tmp
[INF] appending sass content to /Users/lokeshdadoriya/Desktop/Employee_data_management/build/temp/devel
opment/Employee_data_management/sass/Employee_data_management-all.scss.tmp
[INF] appending sass content to /Users/lokeshdadoriya/Desktop/Employee_data_management/build/temp/devel
opment/Employee_data_management/sass/Employee_data_management-all.scss.tmp
[INF] Writing content to /Users/lokeshdadoriya/Desktop/Employee_data_management/build/development/Emplo
yee_data_management/classic.json
[LOG] Fashion build starting for /Users/lokeshdadoriya/Desktop/Employee_data_management/build/temp/deve
lopment/Employee_data_management/sass/Employee_data_management-all.scss
[LOG] Fashion build complete for /Users/lokeshdadoriya/Desktop/Employee_data_management/build/temp/deve
lopment/Employee_data_management/sass/Employee_data_management-all.scss
[LOG] Fashion build completed in 3.977 sec.
[INF] Appending content to /Users/lokeshdadoriya/Desktop/Employee_data_management/bootstrap.js
[INF] Writing content to /Users/lokeshdadoriya/Desktop/Employee_data_management/classic.json
```

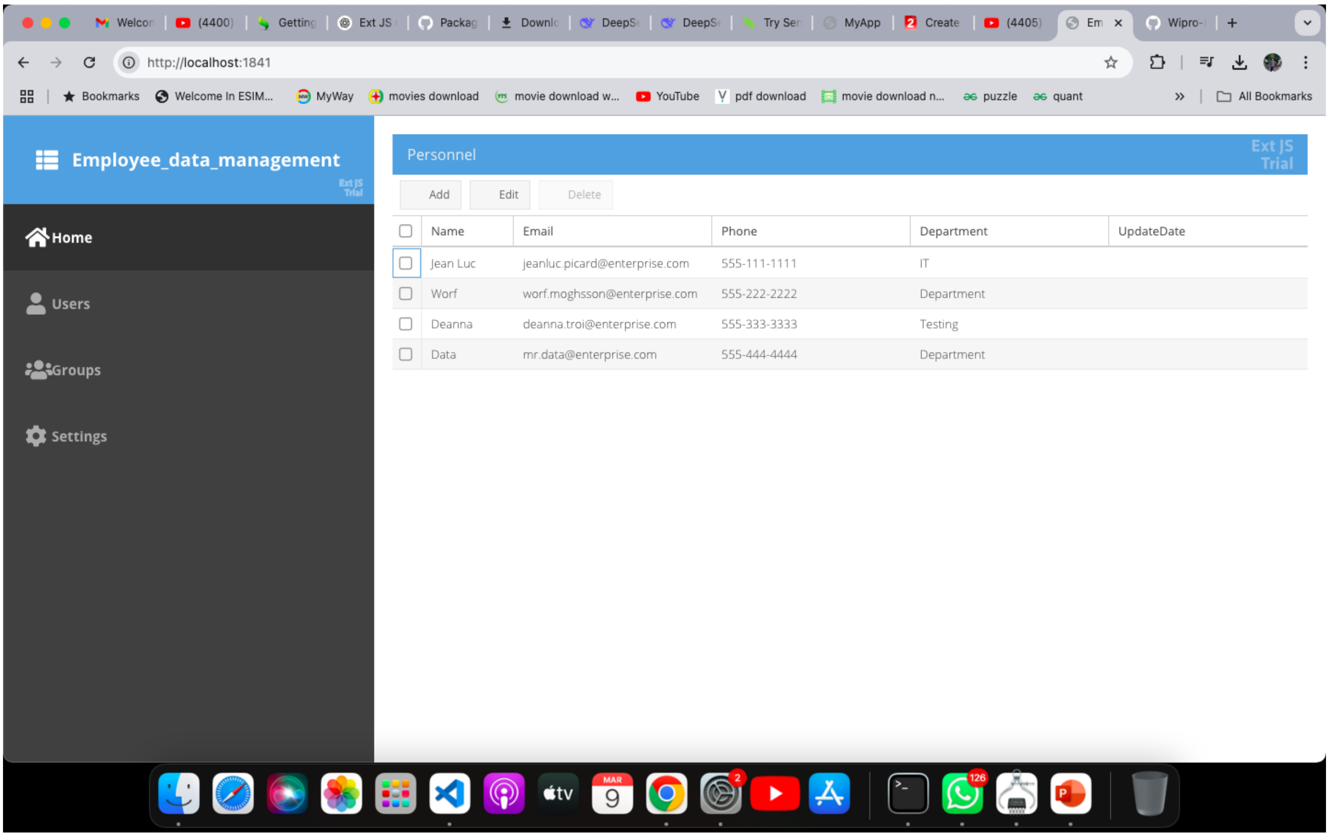
Output:



	Name	Email	Phone	Department	UpdatedDate
□	Jean Luc	jeanluc.picard@enterprise.com	555-111-1111	IT	
□	Worf	worf.moghsson@enterprise.com	555-222-2222	Department	
☑	Deanna	deanna.troi@enterprise.com	555-333-3333	Testing	
□	Data	mr.data@enterprise.com	555-444-4444	Department	



	Name	Email	Phone	Department	UpdateDate
☑	Jean Luc	jeanluc.picard@enterprise.com	555-111-1111	IT	
□	Worf	worf.moghsson@enterprise.com	555-222-2222	Department	
□	Deanna	deanna.troi@enterprise.com	555-333-3333	Testing	
□	Data	mr.data@enterprise.com	555-444-4444	Department	



	Name	Email	Phone	Department	UpdateDate
□	Jean Luc	jeanluc.picard@enterprise.com	555-111-1111	IT	
□	Worf	worf.moghsson@enterprise.com	555-222-2222	Department	
□	Deanna	deanna.troi@enterprise.com	555-333-3333	Testing	
□	Data	mr.data@enterprise.com	555-444-4444	Department	