**KTMOS-MOA**

**Moa-ktmos.com 홈피 에서 로그인을 id:su pw:12345678 접속하면 맨 하단에 환경관리 메뉴가 있습니다.**

**(WEB) 환경관리 – 로그인 이력관리, 다운로드 이력관리 추가**

**환경관리\_다운로드이력관리신규생성(su만접근가능).html**

**환경관리\_로그인이력관리신규생성(su만접근가능).html**

- Description 설명은 ppt 참조하세요 => 파일 : 사용자\_로그인\_다운로드\_이력관리\_20220804.pptx

- 부서부분에 셀렉트박스 부분은 기존에 다른페이지 참조하여 부서부분 동일하게 작업부탁드립니다.

\* Html 파일에는 기본 select로 하드코딩해 놓았습니다.

Css 추가 및 수정건 없습니다.

js부분은 날짜(달력부분), 이력관리(grid부분)

<script type="text/javascript">

    let today = new Date();

    let startDates;

    let role = $('#roleCode').val();

    /\* FU,MU(권한자) : 7일로 변경, AU(권한자) : 3개월로 변경 \*/

    if(role == 'AU'){

        startDates = today.setMonth(today.getMonth() - 3);

    }else if(role == 'FU' || role == 'MU'){

        startDates = today.setDate(today.getDate() - 7);

    }else{

        startDates = today.setMonth(today.getMonth() - 1);

    }

    let picker = tui.DatePicker.createRangePicker({

        language: 'ko',

        startpicker: {

            date: startDates,

            input: '#startDate',

            container: '#datepicker-wrapper1'

        },

        endpicker: {

            date: new Date(),

            input: '#endDate',

            container: '#datepicker-wrapper2'

        }

    });

    let fromDateVal = $('#startDate').val();

    let toDateVal = $('#endDate').val();

    let deviceCategoryVal = $('#deviceCategory').val();

    const dataSource = {

        api: {

            readData: {url: '/additionalfacilities/list/data', method: 'GET',

                initParams: {"startDate": fromDateVal, "endDate": toDateVal, "deviceCategory": deviceCategoryVal}}

        },

        initialRequest: false // 디폴트 값은 true, 아래 gridInit를 통해 호출

    }

</script>

<script type="text/javascript">

    /\*\* TOAST Grid \*/

    const usersGrid2 = new tui.Grid({

        el: document.getElementById('grid2'),

        data: dataSource,

        scrollX: true,

        scrollY: true,

        width: '100%',

        minBodyHeight: 400,

        pageOptions: { perPage: 10 },

        columnOptions: { resizable: true },

        columns: [

            {

                header: '최근접속 일자',

                name: 'Lastaccessdate',

                align: 'center',

                width: 200,

                sortable: true,

                formatter: function ({row, column, value}) {

                    //console.log("1. row: " + JSON.stringify(row) + "\n2. column: " + JSON.stringify(column) + "\n3. value: " + value);

                    let url = '#';

                    return '<a href="' + url + '" style="text-decoration: underline !important; color:#1baec5;">' + value + '</a>';

                }

            },

            {

                header: '구분',

                name: 'userName',

                align: 'left',

                width: 180,

                sortable: true

            },

            {

                header: '아이디',

                name: 'adminid',

                align: 'left',

                width: 180,

                sortable: false

            },

            {

                header: '이름',

                name: 'adminName',

                align: 'left',

                width: 180,

                sortable: false

            },

            {

                header: '부서',

                name: 'department',

                align: 'center',

                width: 340,

                sortable: false

            },

            {

                header: '접근경로',

                name: 'accessroute',

                align: 'center',

                width: 340,

                sortable: false

            },

            {

                header: 'IP',

                name: 'IP',

                align: 'center',

                width: 180,

                sortable: false

            }

        ],

        onGridMounted: function () {

            //console.log("onGridMounted");

        },

        onGridUpdated: function (data) {

            //console.log("onGridUpdated");

            let perPage = data.instance.store.data.pageOptions.perPage;

            let page = data.instance.store.data.pageOptions.page;

            $perPage.val(perPage);

            $page.val(page);

            $lblTotalCount.empty().html(data.instance.store.data.pageOptions.totalCount);

            loading\_mask.close();

        },

        onGridBeforeDestroy: function () {

            //console.log("onGridBeforeDestroy");

        }

    });

</script>

<script type="text/javascript">

    /\*\* TOAST Grid \*/

    const usersGrid3 = new tui.Grid({

        el: document.getElementById('grid3'),

        data: dataSource,

        scrollX: true,

        scrollY: true,

        width: '100%',

        minBodyHeight: 400,

        pageOptions: { perPage: 10 },

        columnOptions: { resizable: true },

        columns: [

            {

                header: '다운로드 일자',

                name: 'datedate',

                align: 'center',

                width: 200,

                sortable: true,

                formatter: function ({row, column, value}) {

                    //console.log("1. row: " + JSON.stringify(row) + "\n2. column: " + JSON.stringify(column) + "\n3. value: " + value);

                    let url = '#';

                    return '<a href="' + url + '" style="text-decoration: underline !important; color:#1baec5;">' + value + '</a>';

                }

            },

            {

                header: '아이디',

                name: 'adminid',

                align: 'left',

                width: 180,

                sortable: false

            },

            {

                header: '이름',

                name: 'adminName',

                align: 'left',

                width: 180,

                sortable: false

            },

            {

                header: '부서',

                name: 'department',

                align: 'center',

                width: 340,

                sortable: false

            },

            {

                header: '접근경로',

                name: 'accessroute',

                align: 'left',

                width: 300,

                sortable: false

            },

            {

                header: '다운로드 파일명',

                name: 'downfile',

                align: 'left',

                width: 340,

                sortable: false

            },

            {

                header: 'IP',

                name: 'IP',

                align: 'center',

                width: 180,

                sortable: false

            }

        ],

        onGridMounted: function () {

            //console.log("onGridMounted");

        },

        onGridUpdated: function (data) {

            //console.log("onGridUpdated");

            let perPage = data.instance.store.data.pageOptions.perPage;

            let page = data.instance.store.data.pageOptions.page;

            $perPage.val(perPage);

            $page.val(page);

            $lblTotalCount.empty().html(data.instance.store.data.pageOptions.totalCount);

            loading\_mask.close();

        },

        onGridBeforeDestroy: function () {

            //console.log("onGridBeforeDestroy");

        }

    });

</script>