E:\workspace\payroll\_rails\app\assets\javascripts\\_payslip.js

//= require\_self

//= require jquery\_lib

//= require jquery.themes.min

//= require utils

//= require admin/wpayslip

E:\workspace\payroll\_rails\app\assets\javascripts\admin.js

// This is a manifest file that'll be compiled into application.js, which will include all the files

// listed below.

//

// Any JavaScript/Coffee file within this directory, lib/assets/javascripts, vendor/assets/javascripts,

// or vendor/assets/javascripts of plugins, if any, can be referenced here using a relative path.

//

// It's not advisable to add code directly here, but if you do, it'll appear at the bottom of the

// the compiled file.

//

// WARNING: THE FIRST BLANK LINE MARKS THE END OF WHAT'S TO BE PROCESSED, ANY BLANK LINE SHOULD

// GO AFTER THE REQUIRES BELOW.

//

//= require application

//= require admin/base

//= require admin/user

//= require admin/emp

//= require admin/designation

//= require admin/empstatus

//= require admin/jobcat

//= require admin/dept

//= require admin/payrate

//= require admin/overtimerate

//= require admin/hpcht

//= require admin/payslip

//= require admin/att

//= require admin/otcht

//= require admin/wkcht

//= require admin/saladj

// require\_tree .

E:\workspace\payroll\_rails\app\assets\javascripts\application.js

// This is a manifest file that'll be compiled into application.js, which will include all the files

// listed below.

//

// Any JavaScript/Coffee file within this directory, lib/assets/javascripts, vendor/assets/javascripts,

// or vendor/assets/javascripts of plugins, if any, can be referenced here using a relative path.

//

// It's not advisable to add code directly here, but if you do, it'll appear at the bottom of the

// the compiled file.

//

// WARNING: THE FIRST BLANK LINE MARKS THE END OF WHAT'S TO BE PROCESSED, ANY BLANK LINE SHOULD

// GO AFTER THE REQUIRES BELOW.

//

//= require\_self

//= require jquery\_lib

//= require jquery\_ujs

//= require jquery\_ui

//= require libs

//= require utils

//= require nav\_list

//= require theme

//= require stat

//= require menu

//= require sort

// require\_tree .

E:\workspace\payroll\_rails\app\assets\javascripts\menu.js

var menu = ( function() {

function menu\_click() {

$('#menu a').removeClass('menu\_active');

$(this).addClass('menu\_active');

}

function get(url, func) {

utils.clear\_dialogs();

$('#contentcolumn div.innertube').load(url, func);

return false;

}

function init() {

$('#menu').accordion({

autoHeight : false,

animated : false

});

$('#menu a').click(menu\_click);

}

return {

get : get,

init : init

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\nav\_list.js

var nav\_list = ( function() {

var config = {

list\_url : '',

list\_func : null,

del\_func : null,

save\_func : null,

search\_param\_func : null

};

/\*\*

\* @public

\* This function shows the list, and reset back the page number to 1.

\*/

function show\_list() {

$('#id\_display').data('pgnum', 1);

utils.stop\_filter\_timer();

update\_list();

}

/\*\*

\* @public

\* This function reloads the list by using the same navigation parameters such as the

\* display size, current page no., and search parameters.

\*/

function update\_list() {

var pgsize = $('#id\_display').val();

var pgnum = $('#id\_display').data('pgnum');

var sort = get\_sort();

var param = ($.isFunction(config.search\_param\_func) ? config.search\_param\_func() : get\_search\_param());

param['pgnum'] = pgnum;

param['pgsize'] = pgsize;

if (sort != null) {

param['sortcolumn'] = sort['column'];

param['sortdir'] = sort['dir'];

}

$.post(config.list\_url, param, function(result) {

$('#right\_box').html(result);

init\_navigate();

});

}

/\*\*

\* @private

\* This function navigate the list to the previous page.

\*/

function go\_prev() {

var val = $('#id\_pg').val();

var arr = val.split(',');

var d = $('#id\_display').val();

$('#id\_display').data('pgnum', arr[2]);

update\_list();

}

/\*\*

\* @private

\* This function navigate the list to the next page.

\*/

function go\_next() {

var val = $('#id\_pg').val();

var arr = val.split(',');

var d = $('#id\_display').val();

$('#id\_display').data('pgnum', arr[3]);

update\_list();

}

/\*\*

\* @private

\* This function navigate the list to the specified page.

\*/

function go\_page() {

var pg = $('#id\_pagenum').val();

var page = 1;

if ($.trim(pg) != '' && $.isNumeric(pg))

page = parseInt(pg);

$('#id\_display').data('pgnum', page);

update\_list();

}

/\*\*

\* @public

\* This function handles the keypress event on the search textbox.

\* It checks for <enter> key.

\*/

function query\_keypress(evt) {

if (evt.keyCode == '13') {

evt.preventDefault();

evt.stopPropagation();

show\_list();

}

}

/\*\*

\* @public

\* This function handles the keyup event on the search textbox.

\* It checks for <enter> key.

\*/

function query\_keyup(evt) {

if (evt.keyCode != '13')

utils.countdown\_filter(show\_list);

}

/\*\*

\* @public

\* This function sets the item text,

\* i.e 1 to 2 of 2

\* @param arg The text.

\*/

function set\_item\_msg(arg) {

$('.item\_display').text(arg);

}

/\*\*

\* @public

\* This function sets the sort info.

\* @param s The sort info in {}.

\*/

function set\_sort(s) {

$('#id\_display').data('sort', s);

update\_list();

}

/\*\*

\* @private

\* This function gets the current sort info in {}.

\* @return The sort info in {}.

\*/

function get\_sort() {

var s = $('#id\_display').data('sort');

return s;

}

/\*\*

\* @private

\* This function initialize the navigation elements after the list has been loaded.

\* The initialization includes :<br>

\* 1. sets the page no. to 1<br>

\* 2. enable/disable the next/prev buttons<br>

\* 3. initialize the edit/delete/clone buttons

\*/

function init\_navigate() {

$('#id\_display').data('pgnum', 1);

if ($.isFunction(config.list\_func))

config.list\_func();

var val = $('#id\_pg').val();

var arr = val.split(',');

if (arr[0] == '0') {

utils.set\_disabled('#id\_prev', 1, null);

}

else {

utils.set\_disabled('#id\_prev', 0, go\_prev);

}

if (arr[1] == '0') {

utils.set\_disabled('#id\_next', 1, null);

}

else {

utils.set\_disabled('#id\_next', 0, go\_next);

}

if (arr[0] == '1' || arr[1] == '1') {

utils.set\_disabled('#id\_go', 0, go\_page);

}

else if (arr[0] == '0' && arr[1] == '0') {

utils.set\_disabled('#id\_go', 1, null);

}

if ($.isFunction(config.del\_func)) {

if ($('.list\_table')[0] != null)

utils.set\_disabled('#id\_delete', 0, config.del\_func);

else

utils.set\_disabled('#id\_delete', 1, null);

}

if ($.isFunction(config.save\_func)) {

if ($('.list\_table')[0] != null)

utils.set\_disabled('#id\_save', 0, config.save\_func);

else

utils.set\_disabled('#id\_save', 1, null);

}

$('#id\_pagenum').val(arr[7]);

sort.init\_sort($('#hd\_' + utils.safe\_replace(arr[5], '.', '-')), arr[6]);

set\_item\_msg(arr[4]);

}

/\*\*

\* @private

\* This function returns the search parameters in {}.

\* @return The search paramaters in {}.

\*/

function get\_search\_param() {

var id\_selection = $('#id\_selection');

var keyword = $('#id\_query').val();

var param = {};

if (id\_selection[0] != null)

param['find'] = id\_selection.val();

if (param['find'] == '0' && keyword == '')

delete param['find']

else

param['keyword'] = keyword;

return param;

}

function init() {

init\_navigate();

}

return {

init : init,

config : config,

show\_list : show\_list,

update\_list : update\_list,

query\_keypress : query\_keypress,

query\_keyup : query\_keyup,

set\_item\_msg : set\_item\_msg,

set\_sort : set\_sort

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\sort.js

var sort = ( function() {

var icon = 'ui-icon';

var icon\_asc = 'ui-icon-triangle-1-n';

var icon\_desc = 'ui-icon-triangle-1-s';

/\*\*

\* @public

\* This function sets the sort icon in the selected column header.

\* @param o The column header selected using $.

\*/

function set\_sort\_css(o) {

var c = o.children().last();

var id = o.attr('id');

id = utils.get\_itemid(id);

var column = utils.safe\_replace(id, '-', '.');

var sort = {

column : column,

dir : 'ASC'

};

var isasc = c.hasClass(icon\_asc);

$('.sorticon').removeClass(icon + ' ' + icon\_asc + ' ' + icon\_desc);

if (isasc)

sort['dir'] = 'DESC';

init\_sort(o, sort.dir);

return sort;

}

/\*\*

\* @public

\* This function sets the corresponding sort icon in the selected column header.

\* @param o The column header selected using $.

\* @param dir The sort direction ASC/DESC.

\*/

function init\_sort(o, dir) {

var c = (dir == 'ASC' ? icon\_asc : icon\_desc);

o.children().last().addClass(icon + ' ' + c);

}

return {

set\_sort\_css : set\_sort\_css,

init\_sort : init\_sort

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\stat.js

var stat = ( function() {

var info = 'ui-state-highlight ui-corner-all';

var info\_icon = 'ui-icon ui-icon-check';

var error = 'ui-state-error ui-corner-all';

var error\_icon = 'ui-icon ui-icon-alert';

var status\_timer = null;

function show\_status(arg, msg) {

clearTimeout(status\_timer);

var id = '#status-panel';

var p = '#status\_msg';

var opt = {};

if (arg == 0) {

set\_info\_class(id);

$(p).html(msg);

$(id).show('drop', opt, 500, function() {

status\_timer = setTimeout(function() {

$(id).hide('scale', opt, 500, null);

}, 5000);

});

}

else {

set\_error\_class(id);

$(p).html(msg);

$(id).show();

}

}

function set\_info\_class(id) {

var div = id + '\_outer';

var span = id + '\_inner';

remove\_status\_class(div, span);

$(div).addClass(info);

$(span).addClass(info\_icon);

}

function set\_error\_class(id) {

var div = id + '\_outer';

var span = id + '\_inner';

remove\_status\_class(div, span);

$(div).addClass(error);

$(span).addClass(error\_icon);

}

function remove\_status\_class(div, span) {

if ($(div).hasClass(info)) {

$(div).removeClass(info);

$(span).removeClass(info\_icon);

}

if ($(div).hasClass(error)) {

$(div).removeClass(error);

$(span).removeClass(error\_icon);

}

}

return {

show\_status : show\_status

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\theme.js

var theme = ( function() {

var default\_theme = 'darkhive';

function init() {

var select\_close = $('#theme\_option .cancelicon');

select\_close.click(function() {

$('#theme\_option').slideUp();

});

$.themes.init({

themes : ['blitzer', 'darkhive', 'trontastic', 'humanity'],

defaultTheme : default\_theme,

onSelect : reload\_IE

});

$('#theme\_body').themes();

$("#main\_options").click(show\_options);

utils.bind\_hover($("#main\_options"));

}

function show\_options() {

$('#theme\_option').slideToggle();

}

function current\_theme() {

return $.themes.currentTheme;

}

function reload\_IE(id, display, url) {

}

return {

init : init,

current\_theme : current\_theme

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\user.js

// This is a manifest file that'll be compiled into application.js, which will include all the files

// listed below.

//

// Any JavaScript/Coffee file within this directory, lib/assets/javascripts, vendor/assets/javascripts,

// or vendor/assets/javascripts of plugins, if any, can be referenced here using a relative path.

//

// It's not advisable to add code directly here, but if you do, it'll appear at the bottom of the

// the compiled file.

//

// WARNING: THE FIRST BLANK LINE MARKS THE END OF WHAT'S TO BE PROCESSED, ANY BLANK LINE SHOULD

// GO AFTER THE REQUIRES BELOW.

//

//= require application

//= require user/base

//= require user/info

//= require user/contact

//= require user/job

//= require user/salary

//= require user/qualification

//= require user/otcht

//= require user/hpcht

//= require user/wkcht

//= require user/payslip

// require\_tree .

E:\workspace\payroll\_rails\app\assets\javascripts\utils.js

var utils = ( function() {

var typing\_timer = null;

var done\_typing\_interval = 2000;

var date\_format = 'dd-mm-yy';

/\*\*

\* @public

\* This function makes the selected id to appear as popup dialog to show alert message.

\* @param id The element id.

\*/

function init\_alert\_dialog(id) {

$(id).dialog({

autoOpen : false,

modal : true,

buttons : {

OK : function() {

$(this).dialog('close');

}

}

});

}

/\*\*

\* @public

\* This function makes the element error-dialog to appear as popup dialog.

\* It also attach the ajaxError event to monitor ajax error.

\*/

function init\_server\_error\_dialog() {

$(document).ajaxError(function(evt, jqXHR, ajaxOptions, errorThrown) {

show\_error\_dialog(jqXHR.responseText);

});

$('#error-dialog').dialog({

autoOpen : false,

modal : true,

width : 700,

height : 500,

buttons : {

OK : function() {

$(this).dialog('close');

}

}

});

}

/\*\*

\* @public

\* This function monitor the progress of an ajax request by showing the progress Loading ...

\*/

function init\_progress() {

$(document).ajaxSend(function(evt, jqXHR, ajaxOptions) {

$('#progress\_status').show();

});

$(document).ajaxComplete(function() {

$('#progress\_status').hide();

});

}

/\*\*

\* @public

\* This function removes a dialog specified by the dialog id.

\* @param id The dialog id.

\*/

function remove\_dialog(id) {

var o = $(id);

try {

o.dialog('destroy');

}

catch (e) {}

o.remove();

}

/\*\*

\* @public

\* This function removes all the dialogs from the DOM.

\*/

function clear\_dialogs() {

remove\_dialog("div[id^='dialog-message']");

remove\_dialog("div[id^='dialog']");

}

/\*\*

\* @public

\* This function shows a message in a popup dialog.

\* @param arg The parameter to determine whether to show html (1) or plain text (2) message.

\* @param msg The message.

\*/

function show\_dialog(arg, msg) {

if (arg == 1)

$('#dialog\_msg').html(msg);

else

$('#dialog\_msg').text(msg);

$('#dialog-message').dialog('open');

}

/\*\*

\* @public

\* This function shows the error message in a popup dialog.

\* @param msg The error message.

\*/

function show\_error\_dialog(msg) {

$('#error\_dialog').html(msg);

$('#error\_dialog style').remove();

$('#error-dialog').dialog('open');

}

/\*\*

\* @public

\* This function binds the hover event to the selected object.

\* @param selector The object selected using $

\*/

function bind\_hover(selector) {

selector.hover(function() {

$(this).toggleClass('ui-state-hover');

}, function() {

$(this).toggleClass('ui-state-hover');

});

}

/\*\*

\* @public

\* This function binds the hover event to the selected list.

\* @param selector The object selected using $

\*/

function bind\_hoverlist(selector) {

selector.hover(function() {

$(this).toggleClass('ui-state-highlight hover');

}, function() {

$(this).toggleClass('ui-state-highlight hover');

});

}

/\*\*

\* @public

\* This function executes the specified function when the typing interval has elapsed.

\* @param func The function to be executed.

\*/

function countdown\_filter(func) {

stop\_filter\_timer();

typing\_timer = setTimeout(func, done\_typing\_interval);

}

/\*\*

\* @public

\* This function stops the typing timer.

\*/

function stop\_filter\_timer() {

clearTimeout(typing\_timer);

}

/\*\*

\* @public

\* This function returns the id from a given element id.

\* e.g if the element id is item\_1,

\* the id will be 1

\* @param arg The element id.

\* @return The id.

\*/

function get\_itemid(arg) {

var i = arg.indexOf('\_');

if (i >= 0) {

var s = arg.substr(i + 1);

return s;

}

return null;

}

/\*\*

\* @public

\* This function enable/disable an element.

\* @param id The element id.

\* @param arg The parameter to enable (0) or disable (1) the element.

\* @param handler The function to be attached to the click event.

\*/

function set\_disabled(id, arg, handler) {

var o = $(id);

o.unbind('click');

o.unbind('mouseenter');

o.unbind('mouseleave');

if (arg == 1) {

o.attr('disabled', 'disabled');

o.removeClass('hover ui-state-hover');

o.addClass('ui-state-disabled');

}

else {

o.removeAttr('disabled');

o.removeClass('ui-state-disabled');

o.addClass('hover');

o.click(handler);

bind\_hover(o);

}

}

/\*\*

\* @public

\* This function replace a string.

\* @param s The string to be replaced.

\* @param a The string to be replaced.

\* @param b The replacement string.

\* @return The replaced string.

\*/

function safe\_replace(s, a, b) {

if (s == null)

return s;

return s.replace(a, b);

}

/\*\*

\* @public

\* This function returns the options for datepicker.

\*/

function date\_opt() {

return {

dateFormat : date\_format,

changeMonth : true,

changeYear : true

};

}

return {

init\_alert\_dialog : init\_alert\_dialog,

init\_server\_error\_dialog : init\_server\_error\_dialog,

init\_progress : init\_progress,

remove\_dialog : remove\_dialog,

clear\_dialogs : clear\_dialogs,

show\_dialog : show\_dialog,

show\_error\_dialog : show\_error\_dialog,

bind\_hover : bind\_hover,

bind\_hoverlist : bind\_hoverlist,

countdown\_filter : countdown\_filter,

stop\_filter\_timer : stop\_filter\_timer,

get\_itemid : get\_itemid,

set\_disabled : set\_disabled,

safe\_replace : safe\_replace,

date\_format : date\_format,

date\_opt : date\_opt

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\att.js

var att = ( function() {

var url = {

list : '/admin/att/list/'

};

function get\_search\_param() {

var param = {

work\_date : $('#id\_work\_date').val(),

employee : $('#id\_employee').val()

};

return param;

}

function sort\_list() {

var s = sort.set\_sort\_css($(this));

nav\_list.set\_sort(s);

}

function init\_list() {

$('.sortheader').click(sort\_list);

}

function init() {

$('.date\_input').datepicker(utils.date\_opt());

$('#id\_find').click(nav\_list.show\_list);

$('#id\_display').change(nav\_list.show\_list);

$('#id\_employee').tooltip({track: true});

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_find'));

nav\_list.config.list\_url = url.list;

nav\_list.config.list\_func = init\_list;

nav\_list.config.search\_param\_func = get\_search\_param;

nav\_list.init();

}

function load() {

return menu.get('/admin/att/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\base.js

var app = ( function() {

function init() {

menu.init();

utils.init\_progress();

utils.init\_server\_error\_dialog();

theme.init();

utils.bind\_hover($('#logout'));

$('#menu\_user').addClass('menu\_active');

user.load();

}

return {

init : init

};

}());

$(app.init);

E:\workspace\payroll\_rails\app\assets\javascripts\admin\dept.js

var dept = ( function() {

var url = {

add : '/admin/dept/new/',

create : '/admin/dept/create/',

edit : '/admin/dept/edit/',

update : '/admin/dept/update/',

del : '/admin/dept/delete/',

list : '/admin/dept/list/'

};

var popup\_dialog\_opt = null;

function init\_ui\_opt() {

popup\_dialog\_opt = {

autoOpen : false,

width : 350,

resizable : false,

draggable : true,

modal : false,

stack : true,

zIndex : 1000

};

}

function show\_form() {

$('#dialog\_add\_body').load(url.add, function() {

$('.save\_button.save').click(func\_save);

$('.save\_button.cancel').click(func\_cancel\_add);

$('#add-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-add').dialog('open');

});

}

function save\_success() {

func\_cancel\_add();

nav\_list.show\_list();

}

function update\_success() {

func\_cancel\_edit();

nav\_list.show\_list();

}

function func\_cancel\_add() {

$('#dialog-add').dialog('close');

return false;

}

function func\_cancel\_edit() {

$('#dialog-edit').dialog('close');

return false;

}

function func\_save() {

var data = get\_data('add');

$('#add-form input').next().remove();

$.post(url.create, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

save\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

$("#add-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_edit(id) {

if (!id)

return false;

id = utils.get\_itemid(id);

$('#dialog\_edit\_body').load(url.edit + id, function() {

$('.save\_button.save').click(function() {

return func\_update(id);

});

$('.save\_button.cancel').click(func\_cancel\_edit);

$('#edit-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-edit').dialog('open');

});

return false;

}

function func\_update(id) {

var data = get\_data('edit');

$('#edit-form input').next().remove();

$.post(url.update + id, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

update\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

$("#edit-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_delete() {

var a = $('.chk:checked');

if (a.length < 1) {

utils.show\_dialog(2, 'Please select record(s).');

return;

}

var l = [];

var trlist = [];

a.each(function(idx, elm) {

var id = $(this).parent().parent().attr('id');

trlist.push('#' + id);

id = utils.get\_itemid(id);

l.push(id);

});

var val = $('#id\_pg').val();

var arr = val.split(',');

var currpg = parseInt(arr[3], 10);

--currpg;

var pgsize = $('#id\_display').val();

var search\_by = $('#id\_selection').val();

var keyword = $('#id\_query').val();

var data = {

'id[]' : l,

pgnum : currpg,

pgsize : pgsize,

find : search\_by,

keyword : keyword

};

$.post(url.del, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

nav\_list.set\_item\_msg(result.itemscount);

var tr = $(trlist.join(','));

tr.remove();

delete tr;

if ($('.list\_table tbody tr').length < 1) {

$('.list\_table').remove();

utils.set\_disabled('#id\_delete', 1, null);

}

}

});

}

function select\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chk').attr('checked', 'checked');

else

$('.chk').removeAttr('checked');

}

function get\_data(t) {

var form = (t == 'add' ? $('#add-form') : $('#edit-form'));

var data = {

name : form.find('#id\_name').val()

};

return data;

}

function sort\_list() {

var s = sort.set\_sort\_css($(this));

nav\_list.set\_sort(s);

}

function init\_list() {

$('.hdchk').click(select\_all);

utils.bind\_hoverlist($('.list\_table tbody tr'));

$('.list\_table tbody').selectable({

selected : function(evt, ui) {

var id = ui.selected.id;

func\_edit(id);

}

});

$('.sortheader').click(sort\_list);

}

function init() {

init\_ui\_opt();

$('#id\_add').click(show\_form);

$('#id\_find').click(nav\_list.show\_list);

$('#id\_display').change(nav\_list.show\_list);

$('#id\_query').keypress(nav\_list.query\_keypress);

$('#id\_query').keyup(nav\_list.query\_keyup);

$('#id\_query').tooltip({track: true});

$('#dialog-add').dialog(popup\_dialog\_opt);

$('#dialog-edit').dialog(popup\_dialog\_opt);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_add,#id\_delete,#id\_find'));

nav\_list.config.list\_url = url.list;

nav\_list.config.list\_func = init\_list;

nav\_list.config.del\_func = func\_delete;

nav\_list.init();

}

function load() {

return menu.get('/admin/dept/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\designation.js

var designation = ( function() {

var url = {

add : '/admin/designation/new/',

create : '/admin/designation/create/',

edit : '/admin/designation/edit/',

update : '/admin/designation/update/',

del : '/admin/designation/delete/',

list : '/admin/designation/list/'

};

var popup\_dialog\_opt = null;

function init\_ui\_opt() {

popup\_dialog\_opt = {

autoOpen : false,

width : 350,

resizable : false,

draggable : true,

modal : false,

stack : true,

zIndex : 1000

};

}

function show\_form() {

$('#dialog\_add\_body').load(url.add, function() {

$('.save\_button.save').click(func\_save);

$('.save\_button.cancel').click(func\_cancel\_add);

$('#add-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-add').dialog('open');

});

}

function save\_success() {

func\_cancel\_add();

nav\_list.show\_list();

}

function update\_success() {

func\_cancel\_edit();

nav\_list.show\_list();

}

function func\_cancel\_add() {

$('#dialog-add').dialog('close');

return false;

}

function func\_cancel\_edit() {

$('#dialog-edit').dialog('close');

return false;

}

function func\_save() {

var data = get\_data('add');

$('#add-form input').next().remove();

$.post(url.create, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

save\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

$("#add-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_edit(id) {

if (!id)

return false;

id = utils.get\_itemid(id);

$('#dialog\_edit\_body').load(url.edit + id, function() {

$('.save\_button.save').click(function() {

return func\_update(id);

});

$('.save\_button.cancel').click(func\_cancel\_edit);

$('#edit-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-edit').dialog('open');

});

return false;

}

function func\_update(id) {

var data = get\_data('edit');

$('#edit-form input').next().remove();

$.post(url.update + id, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

update\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

$("#edit-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_delete() {

var a = $('.chk:checked');

if (a.length < 1) {

utils.show\_dialog(2, 'Please select record(s).');

return;

}

var l = [];

var trlist = [];

a.each(function(idx, elm) {

var id = $(this).parent().parent().attr('id');

trlist.push('#' + id);

id = utils.get\_itemid(id);

l.push(id);

});

var val = $('#id\_pg').val();

var arr = val.split(',');

var currpg = parseInt(arr[3], 10);

--currpg;

var pgsize = $('#id\_display').val();

var search\_by = $('#id\_selection').val();

var keyword = $('#id\_query').val();

var data = {

'id[]' : l,

pgnum : currpg,

pgsize : pgsize,

find : search\_by,

keyword : keyword

};

$.post(url.del, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

nav\_list.set\_item\_msg(result.itemscount);

var tr = $(trlist.join(','));

tr.remove();

delete tr;

if ($('.list\_table tbody tr').length < 1) {

$('.list\_table').remove();

utils.set\_disabled('#id\_delete', 1, null);

}

}

});

}

function select\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chk').attr('checked', 'checked');

else

$('.chk').removeAttr('checked');

}

function get\_data(t) {

var form = (t == 'add' ? $('#add-form') : $('#edit-form'));

var data = {

title : form.find('#id\_title').val(),

desc : form.find('#id\_desc').val(),

note : form.find('#id\_note').val()

};

return data;

}

function sort\_list() {

var s = sort.set\_sort\_css($(this));

nav\_list.set\_sort(s);

}

function init\_list() {

$('.hdchk').click(select\_all);

utils.bind\_hoverlist($('.list\_table tbody tr'));

$('.list\_table tbody').selectable({

selected : function(evt, ui) {

var id = ui.selected.id;

func\_edit(id);

}

});

$('.sortheader').click(sort\_list);

}

function init() {

init\_ui\_opt();

$('#id\_add').click(show\_form);

$('#id\_find').click(nav\_list.show\_list);

$('#id\_display,#id\_selection').change(nav\_list.show\_list);

$('#id\_query').keypress(nav\_list.query\_keypress);

$('#id\_query').keyup(nav\_list.query\_keyup);

$('#id\_query').tooltip({track: true});

$('#dialog-add').dialog(popup\_dialog\_opt);

$('#dialog-edit').dialog(popup\_dialog\_opt);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_add,#id\_delete,#id\_find'));

nav\_list.config.list\_url = url.list;

nav\_list.config.list\_func = init\_list;

nav\_list.config.del\_func = func\_delete;

nav\_list.init();

}

function load() {

return menu.get('/admin/designation/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\emp.js

var emp = ( function() {

var url = {

add : '/admin/employee/new/',

create : '/admin/employee/create/',

edit : '/admin/employee/edit/',

update : '/admin/employee/update/',

del : '/admin/employee/delete/',

list : '/admin/employee/list/'

};

var popup\_dialog\_opt = null;

function init\_ui\_opt() {

popup\_dialog\_opt = {

autoOpen : false,

width : 750,

resizable : false,

draggable : true,

modal : false,

stack : true,

zIndex : 1000

};

}

function show\_form() {

$('#dialog\_add\_body').load(url.add, function() {

$(this).find('#tabs').tabs();

$('.date\_input').datepicker(utils.date\_opt());

$('.save\_button.save').click(func\_save);

$('.save\_button.cancel').click(func\_cancel\_add);

$('#add-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-add').dialog('open');

});

}

function save\_success() {

func\_cancel\_add();

nav\_list.show\_list();

}

function update\_success() {

func\_cancel\_edit();

nav\_list.show\_list();

}

function func\_cancel\_add() {

$('#dialog-add').dialog('close');

return false;

}

function func\_cancel\_edit() {

$('#dialog-edit').dialog('close');

return false;

}

function func\_save() {

var data = get\_data('add');

data = get\_data\_contact('add', data);

data = get\_data\_job('add', data);

data = get\_data\_salary('add', data);

data = get\_data\_qualification('add', data);

$('#add-form input[type="text"]').next().remove();

$('#add-form select').next().remove();

$('#add-form').find('#lb\_gender\_f').next().remove();

$.post(url.create, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

save\_success();

}

else if (result.error == 1) {

if (result.employee.error == 1) {

for (var e in result.employee.errors) {

var d = $('#add-form #form-employee #error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.employee.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

if (e == 'gender')

$('#add-form #form-employee #lb\_gender\_f').after(h);

else if (e == 'marital\_status')

$('#add-form #form-employee #id\_marital\_status').after(h);

else

$("#add-form #form-employee input[name='" + e + "']").after(h);

}

}

}

if (result.employee\_contact.error == 1) {

for (var e in result.employee\_contact.errors) {

var d = $('#add-form #form-employee-contact #error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.employee\_contact.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error\_inline.html',

ext : '.html'

}).render(o);

$("#add-form #form-employee-contact input[name='" + e + "']").after(h);

}

}

}

if (result.employee\_job.error == 1) {

for (var e in result.employee\_job.errors) {

var d = $('#add-form #form-employee-job #error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.employee\_job.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error\_inline.html',

ext : '.html'

}).render(o);

if (e == 'designation\_id' || e == 'department\_id' || e == 'employment\_status\_id' || e == 'job\_category\_id')

$('#add-form #form-employee-job #id\_' + e).after(h);

else

$("#add-form #form-employee-job input[name='" + e + "']").after(h);

}

}

}

if (result.employee\_salary.error == 1) {

for (var e in result.employee\_salary.errors) {

if (!d) {

var o = {

field : e,

msg : result.employee\_salary.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error\_inline.html',

ext : '.html'

}).render(o);

$("#add-form #form-employee-salary input[name='" + e + "']").after(h);

}

}

}

if (result.employee\_qualification.error == 1) {

for (var e in result.employee\_qualification.errors) {

if (!d) {

var o = {

field : e,

msg : result.employee\_qualification.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error\_inline.html',

ext : '.html'

}).render(o);

if (e == 'level')

$('#add-form #form-employee-qualification #id\_' + e).after(h);

else

$("#add-form #form-employee-qualification input[name='" + e + "']").after(h);

}

}

}

utils.show\_dialog(1, 'There are validation errors in the form. Please complete all the required fields.');

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_edit(id) {

if (!id)

return false;

id = utils.get\_itemid(id);

$('#dialog\_edit\_body').load(url.edit + id, function() {

$(this).find('#tabs').tabs();

$('.date\_input').datepicker(utils.date\_opt());

$('.save\_button.save').click(function() {

return func\_update(id);

});

$('.save\_button.cancel').click(func\_cancel\_edit);

$('#edit-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-edit').dialog('open');

});

return false;

}

function func\_update(id) {

var data = get\_data('edit');

data = get\_data\_contact('edit', data);

data = get\_data\_job('edit', data);

data = get\_data\_salary('edit', data);

data = get\_data\_qualification('edit', data);

$('#edit-form input[type="text"]').next().remove();

$('#edit-form select').next().remove();

$('#edit-form').find('#lb\_gender\_f').next().remove();

$.post(url.update + id, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

update\_success();

}

else if (result.error == 1) {

if (result.employee.error == 1) {

for (var e in result.employee.errors) {

var d = $('#edit-form #form-employee #error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.employee.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

if (e == 'gender')

$('#edit-form #form-employee #lb\_gender\_f').after(h);

else if (e == 'marital\_status')

$('#edit-form #form-employee #id\_marital\_status').after(h);

else

$("#edit-form #form-employee input[name='" + e + "']").after(h);

}

}

}

if (result.employee\_contact.error == 1) {

for (var e in result.employee\_contact.errors) {

var d = $('#edit-form #form-employee-contact #error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.employee\_contact.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error\_inline.html',

ext : '.html'

}).render(o);

$("#edit-form #form-employee-contact input[name='" + e + "']").after(h);

}

}

}

if (result.employee\_job.error == 1) {

for (var e in result.employee\_job.errors) {

var d = $('#add-form #form-employee-job #error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.employee\_job.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error\_inline.html',

ext : '.html'

}).render(o);

if (e == 'designation\_id' || e == 'department\_id' || e == 'employment\_status\_id' || e == 'job\_category\_id')

$('#edit-form #form-employee-job #id\_' + e).after(h);

else

$("#edit-form #form-employee-job input[name='" + e + "']").after(h);

}

}

}

if (result.employee\_salary.error == 1) {

for (var e in result.employee\_salary.errors) {

if (!d) {

var o = {

field : e,

msg : result.employee\_salary.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error\_inline.html',

ext : '.html'

}).render(o);

$("#edit-form #form-employee-salary input[name='" + e + "']").after(h);

}

}

}

if (result.employee\_qualification.error == 1) {

for (var e in result.employee\_qualification.errors) {

if (!d) {

var o = {

field : e,

msg : result.employee\_qualification.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error\_inline.html',

ext : '.html'

}).render(o);

if (e == 'level')

$('#edit-form #form-employee-qualification #id\_' + e).after(h);

else

$("#edit-form #form-employee-qualification input[name='" + e + "']").after(h);

}

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_delete() {

var a = $('.chk:checked');

if (a.length < 1) {

utils.show\_dialog(2, 'Please select record(s).');

return;

}

var l = [];

var trlist = [];

a.each(function(idx, elm) {

var id = $(this).parent().parent().attr('id');

trlist.push('#' + id);

id = utils.get\_itemid(id);

l.push(id);

});

var val = $('#id\_pg').val();

var arr = val.split(',');

var currpg = parseInt(arr[3], 10);

--currpg;

var pgsize = $('#id\_display').val();

var data = get\_search\_param();

data['id[]'] = l;

data['pgnum'] = currpg;

data['pgsize'] = pgsize;

$.post(url.del, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

nav\_list.set\_item\_msg(result.itemscount);

var tr = $(trlist.join(','));

tr.remove();

delete tr;

if ($('.list\_table tbody tr')[0] == null) {

$('.list\_table').remove();

utils.set\_disabled('#id\_delete', 1, null);

}

}

});

}

function select\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chk').attr('checked', 'checked');

else

$('.chk').removeAttr('checked');

}

function get\_data(t) {

var form = (t == 'add' ? $('#add-form') : $('#edit-form'));

var data = {

staff\_id : form.find('#id\_staff\_id').val(),

first\_name : form.find('#id\_first\_name').val(),

middle\_name : form.find('#id\_middle\_name').val(),

last\_name : form.find('#id\_last\_name').val(),

new\_ic : form.find('#id\_new\_ic').val(),

old\_ic : form.find('#id\_old\_ic').val(),

passport\_no : form.find('#id\_passport\_no').val(),

gender : form.find("input:radio[name='gender']:checked").val(),

marital\_status : form.find('#id\_marital\_status').val(),

nationality : form.find('#id\_nationality').val(),

dob : form.find('#id\_dob').val(),

place\_of\_birth : form.find('#id\_place\_of\_birth').val(),

race : form.find('#id\_race').val(),

religion : form.find('#id\_religion').val(),

is\_bumi : form.find('#id\_is\_bumi').prop('checked'),

user\_id : form.find('#id\_user\_id').val()

};

return { employee : data };

}

function get\_data\_contact(t, o) {

var f = (t == 'add' ? $('#add-form') : $('#edit-form'));

var form = f.find('#form-employee-contact');

var data = {

address\_1 : form.find('#id\_address\_1').val(),

address\_2 : form.find('#id\_address\_2').val(),

address\_3 : form.find('#id\_address\_3').val(),

city : form.find('#id\_city').val(),

state : form.find('#id\_state').val(),

postcode : form.find('#id\_postcode').val(),

country : form.find('#id\_country').val(),

home\_phone : form.find('#id\_home\_phone').val(),

mobile\_phone : form.find('#id\_mobile\_phone').val(),

work\_email : form.find('#id\_work\_email').val(),

other\_email : form.find('#id\_other\_email').val()

};

o['employee\_contact'] = data;

return o;

}

function get\_data\_job(t, o) {

var f = (t == 'add' ? $('#add-form') : $('#edit-form'));

var form = f.find('#form-employee-job');

var data = {

designation\_id : form.find('#id\_designation\_id').val(),

department\_id : form.find('#id\_department\_id').val(),

employment\_status\_id : form.find('#id\_employment\_status\_id').val(),

job\_category\_id : form.find('#id\_job\_category\_id').val(),

join\_date : form.find('#id\_join\_date').val(),

confirm\_date : form.find('#id\_confirm\_date').val()

};

o['employee\_job'] = data;

return o;

}

function get\_data\_salary(t, o) {

var f = (t == 'add' ? $('#add-form') : $('#edit-form'));

var form = f.find('#form-employee-salary');

var data = {

salary : form.find('#id\_salary').val(),

allowance : form.find('#id\_allowance').val(),

epf : form.find('#id\_epf').val(),

socso : form.find('#id\_socso').val(),

income\_tax : form.find('#id\_income\_tax').val(),

bank\_name : form.find('#id\_bank\_name').val(),

bank\_acc\_no : form.find('#id\_bank\_acc\_no').val(),

bank\_acc\_type : form.find('#id\_bank\_acc\_type').val(),

bank\_address : form.find('#id\_bank\_address').val(),

epf\_no : form.find('#id\_epf\_no').val(),

socso\_no : form.find('#id\_socso\_no').val(),

income\_tax\_no : form.find('#id\_income\_tax\_no').val(),

pay\_type : form.find('#id\_pay\_type').val()

};

o['employee\_salary'] = data;

return o;

}

function get\_data\_qualification(t, o) {

var f = (t == 'add' ? $('#add-form') : $('#edit-form'));

var form = f.find('#form-employee-qualification');

var data = {

level : form.find('#id\_level').val(),

institute : form.find('#id\_institute').val(),

major : form.find('#id\_major').val(),

year : form.find('#id\_year').val(),

gpa : form.find('#id\_gpa').val(),

start\_date : form.find('#id\_start\_date').val(),

end\_date : form.find('#id\_end\_date').val()

};

o['employee\_qualification'] = data;

return o;

}

function get\_search\_param() {

var param = {

employee : $('#id\_employee').val(),

staff\_id : $('#id\_staff\_id').val(),

employment\_status : $('#id\_employment\_status').val(),

designation : $('#id\_designation').val(),

dept : $('#id\_dept').val()

};

return param;

}

function sort\_list() {

var s = sort.set\_sort\_css($(this));

nav\_list.set\_sort(s);

}

function init\_list() {

$('.hdchk').click(select\_all);

utils.bind\_hoverlist($('.list\_table tbody tr'));

$('.list\_table tbody').selectable({

selected : function(evt, ui) {

var id = ui.selected.id;

func\_edit(id);

}

});

$('.sortheader').click(sort\_list);

}

function init() {

init\_ui\_opt();

$('#id\_add').click(show\_form);

$('#id\_find').click(nav\_list.show\_list);

$('#id\_display').change(nav\_list.show\_list);

$('#id\_employee,#id\_staff\_id').tooltip({track : true});

$('#dialog-add').dialog(popup\_dialog\_opt);

$('#dialog-edit').dialog(popup\_dialog\_opt);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_add,#id\_delete,#id\_find'));

nav\_list.config.list\_url = url.list;

nav\_list.config.list\_func = init\_list;

nav\_list.config.del\_func = func\_delete;

nav\_list.config.search\_param\_func = get\_search\_param;

nav\_list.init();

}

function load() {

return menu.get('/admin/employee/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\empstatus.js

var empstatus = ( function() {

var url = {

add : '/admin/empstatus/new/',

create : '/admin/empstatus/create/',

edit : '/admin/empstatus/edit/',

update : '/admin/empstatus/update/',

del : '/admin/empstatus/delete/',

list : '/admin/empstatus/list/'

};

var popup\_dialog\_opt = null;

function init\_ui\_opt() {

popup\_dialog\_opt = {

autoOpen : false,

width : 350,

resizable : false,

draggable : true,

modal : false,

stack : true,

zIndex : 1000

};

}

function show\_form() {

$('#dialog\_add\_body').load(url.add, function() {

$('.save\_button.save').click(func\_save);

$('.save\_button.cancel').click(func\_cancel\_add);

$('#add-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-add').dialog('open');

});

}

function save\_success() {

func\_cancel\_add();

nav\_list.show\_list();

}

function update\_success() {

func\_cancel\_edit();

nav\_list.show\_list();

}

function func\_cancel\_add() {

$('#dialog-add').dialog('close');

return false;

}

function func\_cancel\_edit() {

$('#dialog-edit').dialog('close');

return false;

}

function func\_save() {

var data = get\_data('add');

$('#add-form input').next().remove();

$.post(url.create, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

save\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

$("#add-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_edit(id) {

if (!id)

return false;

id = utils.get\_itemid(id);

$('#dialog\_edit\_body').load(url.edit + id, function() {

$('.save\_button.save').click(function() {

return func\_update(id);

});

$('.save\_button.cancel').click(func\_cancel\_edit);

$('#edit-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-edit').dialog('open');

});

return false;

}

function func\_update(id) {

var data = get\_data('edit');

$('#edit-form input').next().remove();

$.post(url.update + id, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

update\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

$("#edit-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_delete() {

var a = $('.chk:checked');

if (a.length < 1) {

utils.show\_dialog(2, 'Please select record(s).');

return;

}

var l = [];

var trlist = [];

a.each(function(idx, elm) {

var id = $(this).parent().parent().attr('id');

trlist.push('#' + id);

id = utils.get\_itemid(id);

l.push(id);

});

var val = $('#id\_pg').val();

var arr = val.split(',');

var currpg = parseInt(arr[3], 10);

--currpg;

var pgsize = $('#id\_display').val();

var search\_by = $('#id\_selection').val();

var keyword = $('#id\_query').val();

var data = {

'id[]' : l,

pgnum : currpg,

pgsize : pgsize,

find : search\_by,

keyword : keyword

};

$.post(url.del, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

nav\_list.set\_item\_msg(result.itemscount);

var tr = $(trlist.join(','));

tr.remove();

delete tr;

if ($('.list\_table tbody tr').length < 1) {

$('.list\_table').remove();

utils.set\_disabled('#id\_delete', 1, null);

}

}

});

}

function select\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chk').attr('checked', 'checked');

else

$('.chk').removeAttr('checked');

}

function get\_data(t) {

var form = (t == 'add' ? $('#add-form') : $('#edit-form'));

var data = {

name : form.find('#id\_name').val()

};

return data;

}

function sort\_list() {

var s = sort.set\_sort\_css($(this));

nav\_list.set\_sort(s);

}

function init\_list() {

$('.hdchk').click(select\_all);

utils.bind\_hoverlist($('.list\_table tbody tr'));

$('.list\_table tbody').selectable({

selected : function(evt, ui) {

var id = ui.selected.id;

func\_edit(id);

}

});

$('.sortheader').click(sort\_list);

}

function init() {

init\_ui\_opt();

$('#id\_add').click(show\_form);

$('#id\_find').click(nav\_list.show\_list);

$('#id\_display').change(nav\_list.show\_list);

$('#id\_query').keypress(nav\_list.query\_keypress);

$('#id\_query').keyup(nav\_list.query\_keyup);

$('#id\_query').tooltip({track: true});

$('#dialog-add').dialog(popup\_dialog\_opt);

$('#dialog-edit').dialog(popup\_dialog\_opt);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_add,#id\_delete,#id\_find'));

nav\_list.config.list\_url = url.list;

nav\_list.config.list\_func = init\_list;

nav\_list.config.del\_func = func\_delete;

nav\_list.init();

}

function load() {

return menu.get('/admin/empstatus/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\hpcht.js

var hpcht = ( function() {

var url = {

data : '/admin/hourly/chart/data/'

};

// Radialize the colors

Highcharts.getOptions().colors = $.map(Highcharts.getOptions().colors, function(color) {

return {

radialGradient : {

cx : 0.5,

cy : 0.3,

r : 0.7

},

stops : [[0, color], [1, Highcharts.Color(color).brighten(-0.3).get('rgb')] // darken

]

};

});

function chart1(result) {

// Build the chart

var chart = new Highcharts.Chart({

chart : {

renderTo : 'cht1',

plotBackgroundColor : null,

plotBorderWidth : null,

plotShadow : false

},

title : {

text : result.title

},

tooltip : {

pointFormat : '{series.name}: <b>{point.percentage}%</b>',

percentageDecimals : 1

},

plotOptions : {

pie : {

allowPointSelect : true,

cursor : 'pointer',

dataLabels : {

enabled : true,

color : '#000000',

connectorColor : '#000000',

formatter : function() {

return '<b>' + this.point.name + '</b>: ' + this.percentage.toFixed(2) + ' %';

}

}

}

},

series : [{

type : 'pie',

name : result.title,

data : result.pie

}]

});

}

function chart2(result) {

var chart = new Highcharts.Chart({

chart : {

renderTo : 'cht2',

type : 'column'

},

title : {

text : result.title

},

subtitle : {

text : ''

},

xAxis : {

categories : result.column.categories

},

yAxis : {

min : 0,

title : {

text : result.column.yaxis

}

},

legend : {

layout : 'vertical',

backgroundColor : '#FFFFFF',

align : 'left',

verticalAlign : 'top',

x : 100,

y : 70,

floating : true,

shadow : true

},

tooltip : {

formatter : function() {

return '' + this.x + ': ' + this.y;

}

},

plotOptions : {

column : {

pointPadding : 0.2,

borderWidth : 0

}

},

series : [{

name : 'Hourly Payroll',

data : result.column.data,

showInLegend : false

}]

});

}

function draw\_chart() {

var data = get\_search\_param();

$.post(url.data, data, function(result) {

chart1(result);

chart2(result);

});

}

function get\_checked\_month() {

if ($('.chkall').attr('checked') == 'checked')

return 0;

else {

var a = [];

$('.chkmonth:checked').each(function(idx, elm) {

a.push($(this).val());

});

return (a == [] ? 0 : a);

}

}

function get\_search\_param() {

var month = get\_checked\_month();

var param = {

staff\_id : $('#id\_staff\_id').val(),

year : $('#id\_year').val()

};

if (month == 0)

param['month'] = 0;

else

param['month[]'] = month;

return param;

}

function uncheck\_all\_month() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chkmonth').removeAttr('checked');

}

function uncheck\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chkall').removeAttr('checked');

}

function init() {

$('#id\_gen').click(draw\_chart);

$('.chkall').click(uncheck\_all\_month);

$('.chkmonth').click(uncheck\_all);

$('#id\_staff\_id').tooltip({track: true});

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_gen'));

draw\_chart();

}

function load() {

menu.get('/admin/hourly/chart/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\jobcat.js

var jobcat = ( function() {

var url = {

add : '/admin/jobcat/new/',

create : '/admin/jobcat/create/',

edit : '/admin/jobcat/edit/',

update : '/admin/jobcat/update/',

del : '/admin/jobcat/delete/',

list : '/admin/jobcat/list/'

};

var popup\_dialog\_opt = null;

function init\_ui\_opt() {

popup\_dialog\_opt = {

autoOpen : false,

width : 350,

resizable : false,

draggable : true,

modal : false,

stack : true,

zIndex : 1000

};

}

function show\_form() {

$('#dialog\_add\_body').load(url.add, function() {

$('.save\_button.save').click(func\_save);

$('.save\_button.cancel').click(func\_cancel\_add);

$('#add-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-add').dialog('open');

});

}

function save\_success() {

func\_cancel\_add();

nav\_list.show\_list();

}

function update\_success() {

func\_cancel\_edit();

nav\_list.show\_list();

}

function func\_cancel\_add() {

$('#dialog-add').dialog('close');

return false;

}

function func\_cancel\_edit() {

$('#dialog-edit').dialog('close');

return false;

}

function func\_save() {

var data = get\_data('add');

$('#add-form input').next().remove();

$.post(url.create, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

save\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

$("#add-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_edit(id) {

if (!id)

return false;

id = utils.get\_itemid(id);

$('#dialog\_edit\_body').load(url.edit + id, function() {

$('.save\_button.save').click(function() {

return func\_update(id);

});

$('.save\_button.cancel').click(func\_cancel\_edit);

$('#edit-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-edit').dialog('open');

});

return false;

}

function func\_update(id) {

var data = get\_data('edit');

$('#edit-form input').next().remove();

$.post(url.update + id, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

update\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

$("#edit-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_delete() {

var a = $('.chk:checked');

if (a.length < 1) {

utils.show\_dialog(2, 'Please select record(s).');

return;

}

var l = [];

var trlist = [];

a.each(function(idx, elm) {

var id = $(this).parent().parent().attr('id');

trlist.push('#' + id);

id = utils.get\_itemid(id);

l.push(id);

});

var val = $('#id\_pg').val();

var arr = val.split(',');

var currpg = parseInt(arr[3], 10);

--currpg;

var pgsize = $('#id\_display').val();

var search\_by = $('#id\_selection').val();

var keyword = $('#id\_query').val();

var data = {

'id[]' : l,

pgnum : currpg,

pgsize : pgsize,

find : search\_by,

keyword : keyword

};

$.post(url.del, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

nav\_list.set\_item\_msg(result.itemscount);

var tr = $(trlist.join(','));

tr.remove();

delete tr;

if ($('.list\_table tbody tr').length < 1) {

$('.list\_table').remove();

utils.set\_disabled('#id\_delete', 1, null);

}

}

});

}

function select\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chk').attr('checked', 'checked');

else

$('.chk').removeAttr('checked');

}

function get\_data(t) {

var form = (t == 'add' ? $('#add-form') : $('#edit-form'));

var data = {

name : form.find('#id\_name').val()

};

return data;

}

function sort\_list() {

var s = sort.set\_sort\_css($(this));

nav\_list.set\_sort(s);

}

function init\_list() {

$('.hdchk').click(select\_all);

utils.bind\_hoverlist($('.list\_table tbody tr'));

$('.list\_table tbody').selectable({

selected : function(evt, ui) {

var id = ui.selected.id;

func\_edit(id);

}

});

$('.sortheader').click(sort\_list);

}

function init() {

init\_ui\_opt();

$('#id\_add').click(show\_form);

$('#id\_find').click(nav\_list.show\_list);

$('#id\_display').change(nav\_list.show\_list);

$('#id\_query').keypress(nav\_list.query\_keypress);

$('#id\_query').keyup(nav\_list.query\_keyup);

$('#id\_query').tooltip({track: true});

$('#dialog-add').dialog(popup\_dialog\_opt);

$('#dialog-edit').dialog(popup\_dialog\_opt);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_add,#id\_delete,#id\_find'));

nav\_list.config.list\_url = url.list;

nav\_list.config.list\_func = init\_list;

nav\_list.config.del\_func = func\_delete;

nav\_list.init();

}

function load() {

return menu.get('/admin/jobcat/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\otcht.js

var otcht = ( function() {

var url = {

data : '/admin/overtime/chart/data/'

};

function chart1(result) {

var chart = new Highcharts.Chart({

chart : {

renderTo : 'cht1',

type : 'bar'

},

title : {

text : result.title

},

subtitle : {

text : ''

},

xAxis : {

categories : result.categories,

title : {

text : null

}

},

yAxis : {

min : 0,

title : {

text : result.yaxis,

align : 'high'

},

labels : {

overflow : 'justify'

}

},

tooltip : {

formatter : function() {

return '' + this.x + ': ' + this.y + ' hours';

}

},

plotOptions : {

bar : {

dataLabels : {

enabled : true

}

}

},

legend : {

layout : 'vertical',

align : 'right',

verticalAlign : 'top',

x : -100,

y : 100,

floating : true,

borderWidth : 1,

backgroundColor : '#FFFFFF',

shadow : true

},

credits : {

enabled : false

},

series : [{

name : 'Overtime',

data : result.data,

showInLegend : false

}]

});

}

function draw\_chart() {

var data = get\_search\_param();

$.post(url.data, data, function(result) {

chart1(result);

});

}

function get\_checked\_month() {

if ($('.chkall').attr('checked') == 'checked')

return 0;

else {

var a = [];

$('.chkmonth:checked').each(function(idx, elm) {

a.push($(this).val());

});

return (a == [] ? 0 : a);

}

}

function get\_search\_param() {

var month = get\_checked\_month();

var param = {

staff\_id : $('#id\_staff\_id').val(),

year : $('#id\_year').val()

};

if (month == 0)

param['month'] = 0;

else

param['month[]'] = month;

return param;

}

function uncheck\_all\_month() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chkmonth').removeAttr('checked');

}

function uncheck\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chkall').removeAttr('checked');

}

function init() {

$('#id\_gen').click(draw\_chart);

$('.chkall').click(uncheck\_all\_month);

$('.chkmonth').click(uncheck\_all);

$('#id\_staff\_id').tooltip({track: true});

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_gen'));

draw\_chart();

}

function load() {

return menu.get('/admin/overtime/chart/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\overtimerate.js

var overtimerate = ( function() {

var url = {

add : '/admin/overtime/rate/new/',

create : '/admin/overtime/rate/create/',

edit : '/admin/overtime/rate/edit/',

update : '/admin/overtime/rate/update/',

del : '/admin/overtime/rate/delete/',

list : '/admin/overtime/rate/list/'

};

var popup\_dialog\_opt = null;

function init\_ui\_opt() {

popup\_dialog\_opt = {

autoOpen : false,

width : 350,

resizable : false,

draggable : true,

modal : false,

stack : true,

zIndex : 1000

};

}

function show\_form() {

$('#dialog\_add\_body').load(url.add, function() {

$('.save\_button.save').click(func\_save);

$('.save\_button.cancel').click(func\_cancel\_add);

$('#add-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-add').dialog('open');

});

}

function save\_success() {

func\_cancel\_add();

nav\_list.show\_list();

}

function update\_success() {

func\_cancel\_edit();

nav\_list.show\_list();

}

function func\_cancel\_add() {

$('#dialog-add').dialog('close');

return false;

}

function func\_cancel\_edit() {

$('#dialog-edit').dialog('close');

return false;

}

function func\_save() {

var data = get\_data('add');

$('#add-form input').next().remove();

$.post(url.create, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

save\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

if (e == 'year')

$('#add-form #id\_' + e).after(h);

else

$("#add-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_edit(id) {

if (!id)

return false;

id = utils.get\_itemid(id);

$('#dialog\_edit\_body').load(url.edit + id, function() {

$('.save\_button.save').click(function() {

return func\_update(id);

});

$('.save\_button.cancel').click(func\_cancel\_edit);

$('#edit-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-edit').dialog('open');

});

return false;

}

function func\_update(id) {

var data = get\_data('edit');

$('#edit-form input').next().remove();

$.post(url.update + id, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

update\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

if (e == 'year')

$('#edit-form #id\_' + e).after(h);

else

$("#edit-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_delete() {

var a = $('.chk:checked');

if (a.length < 1) {

utils.show\_dialog(2, 'Please select record(s).');

return;

}

var l = [];

var trlist = [];

a.each(function(idx, elm) {

var id = $(this).parent().parent().attr('id');

trlist.push('#' + id);

id = utils.get\_itemid(id);

l.push(id);

});

var val = $('#id\_pg').val();

var arr = val.split(',');

var currpg = parseInt(arr[3], 10);

--currpg;

var pgsize = $('#id\_display').val();

var search\_by = $('#id\_selection').val();

var keyword = $('#id\_query').val();

var data = get\_search\_param();

data['id[]'] = l;

data['pgnum'] = currpg;

data['pgsize'] = pgsize;

$.post(url.del, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

nav\_list.set\_item\_msg(result.itemscount);

var tr = $(trlist.join(','));

tr.remove();

delete tr;

if ($('.list\_table tbody tr')[0] == null) {

$('.list\_table').remove();

utils.set\_disabled('#id\_delete', 1, null);

}

}

});

}

function select\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chk').attr('checked', 'checked');

else

$('.chk').removeAttr('checked');

}

function get\_data(t) {

var form = (t == 'add' ? $('#add-form') : $('#edit-form'));

var data = {

duration : form.find('#id\_duration').val(),

year : form.find('#id\_year').val(),

pay\_rate : form.find('#id\_pay\_rate').val()

};

return data;

}

function get\_search\_param() {

var param = {

year : $('#id\_year').val()

};

return param;

}

function sort\_list() {

var s = sort.set\_sort\_css($(this));

nav\_list.set\_sort(s);

}

function init\_list() {

$('.hdchk').click(select\_all);

utils.bind\_hoverlist($('.list\_table tbody tr'));

$('.list\_table tbody').selectable({

selected : function(evt, ui) {

var id = ui.selected.id;

func\_edit(id);

}

});

$('.sortheader').click(sort\_list);

}

function init() {

init\_ui\_opt();

$('#id\_add').click(show\_form);

$('#id\_find').click(nav\_list.show\_list);

$('#id\_display').change(nav\_list.show\_list);

$('#id\_year').tooltip({track: true});

$('#dialog-add').dialog(popup\_dialog\_opt);

$('#dialog-edit').dialog(popup\_dialog\_opt);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_add,#id\_delete,#id\_find'));

nav\_list.config.list\_url = url.list;

nav\_list.config.list\_func = init\_list;

nav\_list.config.del\_func = func\_delete;

nav\_list.config.search\_param\_func = get\_search\_param;

nav\_list.init();

}

function load() {

return menu.get('/admin/overtime/rate/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\payrate.js

var payrate = ( function() {

var url = {

add : '/admin/payrate/new/',

create : '/admin/payrate/create/',

edit : '/admin/payrate/edit/',

update : '/admin/payrate/update/',

del : '/admin/payrate/delete/',

list : '/admin/payrate/list/'

};

var popup\_dialog\_opt = null;

function init\_ui\_opt() {

popup\_dialog\_opt = {

autoOpen : false,

width : 350,

resizable : false,

draggable : true,

modal : false,

stack : true,

zIndex : 1000

};

}

function show\_form() {

$('#dialog\_add\_body').load(url.add, function() {

$('.save\_button.save').click(func\_save);

$('.save\_button.cancel').click(func\_cancel\_add);

$('#add-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-add').dialog('open');

});

}

function save\_success() {

func\_cancel\_add();

nav\_list.show\_list();

}

function update\_success() {

func\_cancel\_edit();

nav\_list.show\_list();

}

function func\_cancel\_add() {

$('#dialog-add').dialog('close');

return false;

}

function func\_cancel\_edit() {

$('#dialog-edit').dialog('close');

return false;

}

function func\_save() {

var data = get\_data('add');

$('#add-form input').next().remove();

$.post(url.create, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

save\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

if (e == 'month' || e == 'year')

$('#add-form #id\_' + e).after(h);

else

$("#add-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_edit(id) {

if (!id)

return false;

id = utils.get\_itemid(id);

$('#dialog\_edit\_body').load(url.edit + id, function() {

$('.save\_button.save').click(function() {

return func\_update(id);

});

$('.save\_button.cancel').click(func\_cancel\_edit);

$('#edit-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-edit').dialog('open');

});

return false;

}

function func\_update(id) {

var data = get\_data('edit');

$('#edit-form input').next().remove();

$.post(url.update + id, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

update\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

if (e == 'month' || e == 'year')

$('#edit-form #id\_' + e).after(h);

else

$("#edit-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_delete() {

var a = $('.chk:checked');

if (a.length < 1) {

utils.show\_dialog(2, 'Please select record(s).');

return;

}

var l = [];

var trlist = [];

a.each(function(idx, elm) {

var id = $(this).parent().parent().attr('id');

trlist.push('#' + id);

id = utils.get\_itemid(id);

l.push(id);

});

var val = $('#id\_pg').val();

var arr = val.split(',');

var currpg = parseInt(arr[3], 10);

--currpg;

var pgsize = $('#id\_display').val();

var search\_by = $('#id\_selection').val();

var keyword = $('#id\_query').val();

var data = get\_search\_param();

data['id[]'] = l;

data['pgnum'] = currpg;

data['pgsize'] = pgsize;

$.post(url.del, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

nav\_list.set\_item\_msg(result.itemscount);

var tr = $(trlist.join(','));

tr.remove();

delete tr;

if ($('.list\_table tbody tr')[0] == null) {

$('.list\_table').remove();

utils.set\_disabled('#id\_delete', 1, null);

}

}

});

}

function select\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chk').attr('checked', 'checked');

else

$('.chk').removeAttr('checked');

}

function get\_data(t) {

var form = (t == 'add' ? $('#add-form') : $('#edit-form'));

var data = {

staff\_id : form.find('#id\_staff\_id').val(),

month : form.find('#id\_month').val(),

year : form.find('#id\_year').val(),

pay\_rate : form.find('#id\_pay\_rate').val()

};

return data;

}

function get\_search\_param() {

var param = {

staff\_id : $('#id\_staff\_id').val(),

month : $('#id\_month').val(),

year : $('#id\_year').val()

};

return param;

}

function sort\_list() {

var s = sort.set\_sort\_css($(this));

nav\_list.set\_sort(s);

}

function init\_list() {

$('.hdchk').click(select\_all);

utils.bind\_hoverlist($('.list\_table tbody tr'));

$('.list\_table tbody').selectable({

selected : function(evt, ui) {

var id = ui.selected.id;

func\_edit(id);

}

});

$('.sortheader').click(sort\_list);

}

function init() {

init\_ui\_opt();

$('#id\_add').click(show\_form);

$('#id\_find').click(nav\_list.show\_list);

$('#id\_display').change(nav\_list.show\_list);

$('#id\_staff\_id').tooltip({track: true});

$('#dialog-add').dialog(popup\_dialog\_opt);

$('#dialog-edit').dialog(popup\_dialog\_opt);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_add,#id\_delete,#id\_find'));

nav\_list.config.list\_url = url.list;

nav\_list.config.list\_func = init\_list;

nav\_list.config.del\_func = func\_delete;

nav\_list.config.search\_param\_func = get\_search\_param;

nav\_list.init();

}

function load() {

return menu.get('/admin/payrate/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\payslip.js

var payslip = ( function() {

var url = {

list : '/admin/payslip/list/',

salaryslip : '/admin/payslip/slip/'

};

function func\_generate(id) {

if (!id)

return false;

id = utils.get\_itemid(id);

var month = $('#id\_month').val();

var year = $('#id\_year').val();

var f = 'width=800,height=400,menubar=1';

open(url.salaryslip + id + '/' + month + '/' + year, '\_blank', f);

return false;

}

function get\_search\_param() {

var param = {

employee : $('#id\_employee').val(),

staff\_id : $('#id\_staff\_id').val(),

employment\_status : $('#id\_employment\_status').val(),

designation : $('#id\_designation').val(),

dept : $('#id\_dept').val()

};

return param;

}

function sort\_list() {

var s = sort.set\_sort\_css($(this));

nav\_list.set\_sort(s);

}

function init\_list() {

utils.bind\_hoverlist($('.list\_table tbody tr'));

$('.list\_table tbody').selectable({

selected : function(evt, ui) {

var id = ui.selected.id;

func\_generate(id);

}

});

$('.sortheader').click(sort\_list);

}

function init() {

$('#id\_find').click(nav\_list.show\_list);

$('#id\_display').change(nav\_list.show\_list);

$('#id\_employee,#id\_staff\_id').tooltip({track : true});

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_find'));

nav\_list.config.list\_url = url.list;

nav\_list.config.list\_func = init\_list;

nav\_list.config.search\_param\_func = get\_search\_param;

nav\_list.init();

}

function load() {

return menu.get('/admin/payslip/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\saladj.js

var saladj = ( function() {

var url = {

add : '/admin/salaryadj/new/',

create : '/admin/salaryadj/create/',

edit : '/admin/salaryadj/edit/',

update : '/admin/salaryadj/update/',

del : '/admin/salaryadj/delete/',

list : '/admin/salaryadj/list/'

};

var popup\_dialog\_opt = null;

function init\_ui\_opt() {

popup\_dialog\_opt = {

autoOpen : false,

width : 350,

resizable : false,

draggable : true,

modal : false,

stack : true,

zIndex : 1000

};

}

function show\_form() {

$('#dialog\_add\_body').load(url.add, function() {

$('.save\_button.save').click(func\_save);

$('.save\_button.cancel').click(func\_cancel\_add);

$('#add-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-add').dialog('open');

});

}

function save\_success() {

func\_cancel\_add();

nav\_list.show\_list();

}

function update\_success() {

func\_cancel\_edit();

nav\_list.show\_list();

}

function func\_cancel\_add() {

$('#dialog-add').dialog('close');

return false;

}

function func\_cancel\_edit() {

$('#dialog-edit').dialog('close');

return false;

}

function func\_save() {

var data = get\_data('add');

$('#add-form input').next().remove();

$.post(url.create, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

save\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

if (e == 'month' || e == 'year')

$('#add-form #id\_' + e).after(h);

else

$("#add-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_edit(id) {

if (!id)

return false;

id = utils.get\_itemid(id);

$('#dialog\_edit\_body').load(url.edit + id, function() {

$('.save\_button.save').click(function() {

return func\_update(id);

});

$('.save\_button.cancel').click(func\_cancel\_edit);

$('#edit-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-edit').dialog('open');

});

return false;

}

function func\_update(id) {

var data = get\_data('edit');

$('#edit-form input').next().remove();

$.post(url.update + id, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

update\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

if (e == 'month' || e == 'year')

$('#edit-form #id\_' + e).after(h);

else

$("#edit-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_delete() {

var a = $('.chk:checked');

if (a.length < 1) {

utils.show\_dialog(2, 'Please select record(s).');

return;

}

var l = [];

var trlist = [];

a.each(function(idx, elm) {

var id = $(this).parent().parent().attr('id');

trlist.push('#' + id);

id = utils.get\_itemid(id);

l.push(id);

});

var val = $('#id\_pg').val();

var arr = val.split(',');

var currpg = parseInt(arr[3], 10);

--currpg;

var pgsize = $('#id\_display').val();

var search\_by = $('#id\_selection').val();

var keyword = $('#id\_query').val();

var data = get\_search\_param();

data['id[]'] = l;

data['pgnum'] = currpg;

data['pgsize'] = pgsize;

$.post(url.del, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

nav\_list.set\_item\_msg(result.itemscount);

var tr = $(trlist.join(','));

tr.remove();

delete tr;

if ($('.list\_table tbody tr')[0] == null) {

$('.list\_table').remove();

utils.set\_disabled('#id\_delete', 1, null);

}

}

});

}

function select\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chk').attr('checked', 'checked');

else

$('.chk').removeAttr('checked');

}

function get\_data(t) {

var form = (t == 'add' ? $('#add-form') : $('#edit-form'));

var data = {

staff\_id : form.find('#id\_staff\_id').val(),

inc : form.find('#id\_inc').val(),

month : form.find('#id\_month').val(),

year : form.find('#id\_year').val()

};

return data;

}

function get\_search\_param() {

var param = {

staff\_id : $('#id\_staff\_id').val(),

month : $('#id\_month').val(),

year : $('#id\_year').val()

};

return param;

}

function sort\_list() {

var s = sort.set\_sort\_css($(this));

nav\_list.set\_sort(s);

}

function init\_list() {

$('.hdchk').click(select\_all);

utils.bind\_hoverlist($('.list\_table tbody tr'));

$('.list\_table tbody').selectable({

selected : function(evt, ui) {

var id = ui.selected.id;

func\_edit(id);

}

});

$('.sortheader').click(sort\_list);

}

function init() {

init\_ui\_opt();

$('#id\_add').click(show\_form);

$('#id\_find').click(nav\_list.show\_list);

$('#id\_display').change(nav\_list.show\_list);

$('#id\_staff\_id').tooltip({track: true});

$('#dialog-add').dialog(popup\_dialog\_opt);

$('#dialog-edit').dialog(popup\_dialog\_opt);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_add,#id\_delete,#id\_find'));

nav\_list.config.list\_url = url.list;

nav\_list.config.list\_func = init\_list;

nav\_list.config.del\_func = func\_delete;

nav\_list.config.search\_param\_func = get\_search\_param;

nav\_list.init();

}

function load() {

return menu.get('/admin/salaryadj/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\user.js

var user = ( function() {

var url = {

add : '/admin/user/new/',

create : '/admin/user/create/',

edit : '/admin/user/edit/',

update : '/admin/user/update/',

del : '/admin/user/delete/',

list : '/admin/user/list/'

};

var popup\_dialog\_opt = null;

function init\_ui\_opt() {

popup\_dialog\_opt = {

autoOpen : false,

width : 350,

resizable : false,

draggable : true,

modal : false,

stack : true,

zIndex : 1000

};

}

function show\_form() {

$('#dialog\_add\_body').load(url.add, function() {

$('.save\_button.save').click(func\_save);

$('.save\_button.cancel').click(func\_cancel\_add);

$('#add-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-add').dialog('open');

});

}

function save\_success() {

func\_cancel\_add();

nav\_list.show\_list();

}

function update\_success() {

func\_cancel\_edit();

nav\_list.show\_list();

}

function func\_cancel\_add() {

$('#dialog-add').dialog('close');

return false;

}

function func\_cancel\_edit() {

$('#dialog-edit').dialog('close');

return false;

}

function func\_save() {

var data = get\_data('add');

$('#add-form input').next().remove();

$.post(url.create, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

save\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

$("#add-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_edit(id) {

if (!id)

return false;

id = utils.get\_itemid(id);

$('#dialog\_edit\_body').load(url.edit + id, function() {

$('#edit-form').find('#id\_password').val('\*\*\*\*\*\*\*\*');

$('.save\_button.save').click(function() {

return func\_update(id);

});

$('.save\_button.cancel').click(func\_cancel\_edit);

$('#edit-form').tooltip({track: true});

utils.bind\_hover($('.save\_button'));

$('#dialog-edit').dialog('open');

});

return false;

}

function func\_update(id) {

var data = get\_data('edit');

$('#edit-form input').next().remove();

$.post(url.update + id, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

update\_success();

}

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

$("#edit-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function func\_delete() {

var a = $('.chk:checked');

if (a.length < 1) {

utils.show\_dialog(2, 'Please select record(s).');

return;

}

var l = [];

var trlist = [];

a.each(function(idx, elm) {

var id = $(this).parent().parent().attr('id');

trlist.push('#' + id);

id = utils.get\_itemid(id);

l.push(id);

});

var val = $('#id\_pg').val();

var arr = val.split(',');

var currpg = parseInt(arr[3], 10);

--currpg;

var pgsize = $('#id\_display').val();

var search\_by = $('#id\_selection').val();

var keyword = $('#id\_query').val();

var data = get\_search\_param();

data['id[]'] = l;

data['pgnum'] = currpg;

data['pgsize'] = pgsize;

$.post(url.del, data, function(result) {

if (result.success == 1) {

stat.show\_status(0, result.message);

nav\_list.set\_item\_msg(result.itemscount);

var tr = $(trlist.join(','));

tr.remove();

delete tr;

if ($('.list\_table tbody tr')[0] == null) {

$('.list\_table').remove();

utils.set\_disabled('#id\_delete', 1, null);

}

}

});

}

function select\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chk').attr('checked', 'checked');

else

$('.chk').removeAttr('checked');

}

function get\_data(t) {

var form = (t == 'add' ? $('#add-form') : $('#edit-form'));

var data = {

role : form.find('#id\_role').val(),

username : form.find('#id\_username').val(),

status : form.find('#id\_status').val(),

pwd : form.find('#id\_password').val(),

pwdconfirm : form.find('#id\_passwordconfirm').val()

};

return data;

}

function get\_search\_param() {

var param = {

username : $('#id\_username').val(),

role : $('#id\_user\_role').val(),

employee : $('#id\_employee').val(),

status : $('#id\_status').val()

};

return param;

}

function sort\_list() {

var s = sort.set\_sort\_css($(this));

nav\_list.set\_sort(s);

}

function init\_list() {

$('.hdchk').click(select\_all);

utils.bind\_hoverlist($('.list\_table tbody tr'));

$('.list\_table tbody').selectable({

selected : function(evt, ui) {

var id = ui.selected.id;

func\_edit(id);

}

});

$('.sortheader').click(sort\_list);

}

function init() {

init\_ui\_opt();

$('#id\_add').click(show\_form);

$('#id\_find').click(nav\_list.show\_list);

$('#id\_display').change(nav\_list.show\_list);

$('#id\_username,#id\_employee').tooltip({track: true});

$('#dialog-add').dialog(popup\_dialog\_opt);

$('#dialog-edit').dialog(popup\_dialog\_opt);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_add,#id\_delete,#id\_find'));

nav\_list.config.list\_url = url.list;

nav\_list.config.list\_func = init\_list;

nav\_list.config.del\_func = func\_delete;

nav\_list.config.search\_param\_func = get\_search\_param;

nav\_list.init();

}

function load() {

return menu.get('/admin/user/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\wkcht.js

var wkcht = ( function() {

var url = {

data : '/admin/workhours/chart/data/'

}

function chart1(result) {

var chart = new Highcharts.Chart({

chart : {

renderTo : 'cht1',

type : 'line'

},

title : {

text : result.title

},

subtitle : {

text : ''

},

xAxis : {

categories : result.categories

},

yAxis : {

title : {

text : result.yaxis

},

plotLines : [{

value : 0,

width : 1,

color : '#808080'

}]

},

tooltip : {

formatter : function() {

return '' + this.x + ': ' + this.y + ' hours';

}

},

plotOptions : {

line : {

dataLabels : {

enabled : true

}

}

},

legend : {

layout : 'vertical',

align : 'right',

verticalAlign : 'top',

x : -10,

y : 100,

borderWidth : 0

},

series : [{

name : 'Hours Worked',

data : result.data,

showInLegend : false

}]

});

}

function draw\_chart() {

var data = get\_search\_param();

$.post(url.data, data, function(result) {

chart1(result);

});

}

function get\_checked\_month() {

if ($('.chkall').attr('checked') == 'checked')

return 0;

else {

var a = [];

$('.chkmonth:checked').each(function(idx, elm) {

a.push($(this).val());

});

return (a == [] ? 0 : a);

}

}

function get\_search\_param() {

var month = get\_checked\_month();

var param = {

staff\_id : $('#id\_staff\_id').val(),

year : $('#id\_year').val()

};

if (month == 0)

param['month'] = 0;

else

param['month[]'] = month;

return param;

}

function uncheck\_all\_month() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chkmonth').removeAttr('checked');

}

function uncheck\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chkall').removeAttr('checked');

}

function init() {

$('#id\_gen').click(draw\_chart);

$('.chkall').click(uncheck\_all\_month);

$('.chkmonth').click(uncheck\_all);

$('#id\_staff\_id').tooltip({track: true});

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_gen'));

draw\_chart();

}

function load() {

return menu.get('/admin/workhours/chart/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\admin\wpayslip.js

var wpayslip = ( function() {

function init() {

utils.bind\_hover($('.print\_button'));

$('.print\_button').click(function() {

print();

});

var w = opener;

var \_theme = w.theme.current\_theme();

$.themes.init({defaultTheme : \_theme});

}

return {

init : init

};

}());

$(document).ready(wpayslip.init);

E:\workspace\payroll\_rails\app\assets\javascripts\user\base.js

var app = ( function() {

function init() {

menu.init();

utils.init\_progress();

utils.init\_server\_error\_dialog();

theme.init();

utils.bind\_hover($('#logout'));

$('#menu\_info').addClass('menu\_active');

info.load();

}

return {

init : init

};

}());

$(app.init);

E:\workspace\payroll\_rails\app\assets\javascripts\user\contact.js

var contact = ( function() {

var url = {

update : '/user/contact/update/'

};

function func\_save() {

var data = get\_data();

$('#save-form input[type="text"]').next().remove();

$.post(url.update, data, function(result) {

if (result.success == 1)

stat.show\_status(0, result.message);

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#save-form #error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error\_inline.html',

ext : '.html'

}).render(o);

$("#save-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function get\_data() {

var form = $('#save-form');

var data = {

address\_1 : form.find('#id\_address\_1').val(),

address\_2 : form.find('#id\_address\_2').val(),

address\_3 : form.find('#id\_address\_3').val(),

city : form.find('#id\_city').val(),

state : form.find('#id\_state').val(),

postcode : form.find('#id\_postcode').val(),

country : form.find('#id\_country').val(),

home\_phone : form.find('#id\_home\_phone').val(),

mobile\_phone : form.find('#id\_mobile\_phone').val(),

work\_email : form.find('#id\_work\_email').val(),

other\_email : form.find('#id\_other\_email').val()

};

return {

employee\_contact : data

};

}

function init() {

$('.save\_button.save').click(func\_save);

$('#save-form').tooltip({track : true});

utils.bind\_hover($('.save\_button'));

utils.init\_alert\_dialog('#dialog-message');

}

function load() {

return menu.get('/user/contact/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\user\hpcht.js

var hpcht = ( function() {

var url = {

data : '/user/hourly/chart/data/'

};

// Radialize the colors

Highcharts.getOptions().colors = $.map(Highcharts.getOptions().colors, function(color) {

return {

radialGradient : {

cx : 0.5,

cy : 0.3,

r : 0.7

},

stops : [[0, color], [1, Highcharts.Color(color).brighten(-0.3).get('rgb')] // darken

]

};

});

function chart1(result) {

// Build the chart

var chart = new Highcharts.Chart({

chart : {

renderTo : 'cht1',

plotBackgroundColor : null,

plotBorderWidth : null,

plotShadow : false

},

title : {

text : result.title

},

tooltip : {

pointFormat : '{series.name}: <b>{point.percentage}%</b>',

percentageDecimals : 1

},

plotOptions : {

pie : {

allowPointSelect : true,

cursor : 'pointer',

dataLabels : {

enabled : true,

color : '#000000',

connectorColor : '#000000',

formatter : function() {

return '<b>' + this.point.name + '</b>: ' + this.percentage.toFixed(2) + ' %';

}

}

}

},

series : [{

type : 'pie',

name : result.title,

data : result.pie

}]

});

}

function chart2(result) {

var chart = new Highcharts.Chart({

chart : {

renderTo : 'cht2',

type : 'column'

},

title : {

text : result.title

},

subtitle : {

text : ''

},

xAxis : {

categories : result.column.categories

},

yAxis : {

min : 0,

title : {

text : result.column.yaxis

}

},

legend : {

layout : 'vertical',

backgroundColor : '#FFFFFF',

align : 'left',

verticalAlign : 'top',

x : 100,

y : 70,

floating : true,

shadow : true

},

tooltip : {

formatter : function() {

return '' + this.x + ': ' + this.y;

}

},

plotOptions : {

column : {

pointPadding : 0.2,

borderWidth : 0

}

},

series : [{

name : 'Hourly Payroll',

data : result.column.data,

showInLegend : false

}]

});

}

function draw\_chart() {

var data = get\_search\_param();

$.post(url.data, data, function(result) {

chart1(result);

chart2(result);

});

}

function get\_checked\_month() {

if ($('.chkall').attr('checked') == 'checked')

return 0;

else {

var a = [];

$('.chkmonth:checked').each(function(idx, elm) {

a.push($(this).val());

});

return (a == [] ? 0 : a);

}

}

function get\_search\_param() {

var month = get\_checked\_month();

var param = {

year : $('#id\_year').val()

};

if (month == 0)

param['month'] = 0;

else

param['month[]'] = month;

return param;

}

function uncheck\_all\_month() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chkmonth').removeAttr('checked');

}

function uncheck\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chkall').removeAttr('checked');

}

function init() {

$('#id\_gen').click(draw\_chart);

$('.chkall').click(uncheck\_all\_month);

$('.chkmonth').click(uncheck\_all);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_gen'));

draw\_chart();

}

function load() {

menu.get('/user/hourly/chart/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\user\info.js

var info = ( function() {

var url = {

update : '/user/info/update/'

};

function func\_save() {

var data = get\_data();

$('#save-form input[type="text"]').next().remove();

$('#save-form select').next().remove();

$('#save-form').find('#lb\_gender\_f').next().remove();

$.post(url.update, data, function(result) {

if (result.success == 1)

stat.show\_status(0, result.message);

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#save-form #error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error.html',

ext : '.html'

}).render(o);

if (e == 'gender')

$('#save-form #lb\_gender\_f').after(h);

else if (e == 'marital\_status')

$('#save-form #id\_marital\_status').after(h);

else

$("#save-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function get\_data() {

var form = $('#save-form');

var data = {

first\_name : form.find('#id\_first\_name').val(),

middle\_name : form.find('#id\_middle\_name').val(),

last\_name : form.find('#id\_last\_name').val(),

new\_ic : form.find('#id\_new\_ic').val(),

old\_ic : form.find('#id\_old\_ic').val(),

passport\_no : form.find('#id\_passport\_no').val(),

gender : form.find("input:radio[name='gender']:checked").val(),

marital\_status : form.find('#id\_marital\_status').val(),

nationality : form.find('#id\_nationality').val(),

dob : form.find('#id\_dob').val(),

place\_of\_birth : form.find('#id\_place\_of\_birth').val(),

race : form.find('#id\_race').val(),

religion : form.find('#id\_religion').val(),

is\_bumi : form.find('#id\_is\_bumi').prop('checked')

};

return {

employee : data

};

}

function init() {

$('.date\_input').datepicker(utils.date\_opt());

$('.save\_button.save').click(func\_save);

$('#save-form').tooltip({track : true});

utils.bind\_hover($('.save\_button'));

utils.init\_alert\_dialog('#dialog-message');

}

function load() {

return menu.get('/user/info/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\user\job.js

var job = ( function() {

function init() {

utils.init\_alert\_dialog('#dialog-message');

}

function load() {

return menu.get('/user/job/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\user\otcht.js

var otcht = ( function() {

var url = {

data : '/user/overtime/chart/data/'

};

function chart1(result) {

var chart = new Highcharts.Chart({

chart : {

renderTo : 'cht1',

type : 'bar'

},

title : {

text : result.title

},

subtitle : {

text : ''

},

xAxis : {

categories : result.categories,

title : {

text : null

}

},

yAxis : {

min : 0,

title : {

text : result.yaxis,

align : 'high'

},

labels : {

overflow : 'justify'

}

},

tooltip : {

formatter : function() {

return '' + this.x + ': ' + this.y + ' hours';

}

},

plotOptions : {

bar : {

dataLabels : {

enabled : true

}

}

},

legend : {

layout : 'vertical',

align : 'right',

verticalAlign : 'top',

x : -100,

y : 100,

floating : true,

borderWidth : 1,

backgroundColor : '#FFFFFF',

shadow : true

},

credits : {

enabled : false

},

series : [{

name : 'Overtime',

data : result.data,

showInLegend : false

}]

});

}

function draw\_chart() {

var data = get\_search\_param();

$.post(url.data, data, function(result) {

chart1(result);

});

}

function get\_checked\_month() {

if ($('.chkall').attr('checked') == 'checked')

return 0;

else {

var a = [];

$('.chkmonth:checked').each(function(idx, elm) {

a.push($(this).val());

});

return (a == [] ? 0 : a);

}

}

function get\_search\_param() {

var month = get\_checked\_month();

var param = {

year : $('#id\_year').val()

};

if (month == 0)

param['month'] = 0;

else

param['month[]'] = month;

return param;

}

function uncheck\_all\_month() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chkmonth').removeAttr('checked');

}

function uncheck\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chkall').removeAttr('checked');

}

function init() {

$('#id\_gen').click(draw\_chart);

$('.chkall').click(uncheck\_all\_month);

$('.chkmonth').click(uncheck\_all);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_gen'));

draw\_chart();

}

function load() {

return menu.get('/user/overtime/chart/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\user\payslip.js

var payslip = ( function() {

var url = {

salaryslip : '/user/payslip/slip/'

};

function func\_generate() {

var month = $('#id\_month').val();

var year = $('#id\_year').val();

var f = 'width=800,height=400,menubar=1';

open(url.salaryslip + month + '/' + year, '\_blank', f);

return false;

}

function init() {

$('#id\_gen').click(func\_generate);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_gen'));

}

function load() {

return menu.get('/user/payslip/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\user\qualification.js

var qualification = ( function() {

var url = {

update : '/user/qualification/update/'

};

function func\_save() {

var data = get\_data();

$('#save-form input[type="text"]').next().remove();

$('#save-form select').next().remove();

$.post(url.update, data, function(result) {

if (result.success == 1)

stat.show\_status(0, result.message);

else if (result.error == 1) {

for (var e in result.errors) {

var d = $('#save-form #error\_' + e).get(0);

if (!d) {

var o = {

field : e,

msg : result.errors[e][0]

};

var h = new EJS({

url : '/assets/tpl/label\_error\_inline.html',

ext : '.html'

}).render(o);

$("#save-form input[name='" + e + "']").after(h);

}

}

}

else

utils.show\_dialog(2, result);

});

return false;

}

function get\_data() {

var form = $('#save-form');

var data = {

level : form.find('#id\_level').val(),

institute : form.find('#id\_institute').val(),

major : form.find('#id\_major').val(),

year : form.find('#id\_year').val(),

gpa : form.find('#id\_gpa').val(),

start\_date : form.find('#id\_start\_date').val(),

end\_date : form.find('#id\_end\_date').val()

};

return {

employee\_qualification : data

};

}

function init() {

$('.date\_input').datepicker(utils.date\_opt());

$('.save\_button.save').click(func\_save);

$('#save-form').tooltip({track : true});

utils.bind\_hover($('.save\_button'));

utils.init\_alert\_dialog('#dialog-message');

}

function load() {

return menu.get('/user/qualification/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\user\salary.js

var salary = ( function() {

function init() {

utils.init\_alert\_dialog('#dialog-message');

}

function load() {

return menu.get('/user/salary/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\user\wkcht.js

var wkcht = ( function() {

var url = {

data : '/user/workhours/chart/data/'

};

function chart1(result) {

var chart = new Highcharts.Chart({

chart : {

renderTo : 'cht1',

type : 'line'

},

title : {

text : result.title

},

subtitle : {

text : ''

},

xAxis : {

categories : result.categories

},

yAxis : {

title : {

text : result.yaxis

},

plotLines : [{

value : 0,

width : 1,

color : '#808080'

}]

},

tooltip : {

formatter : function() {

return '' + this.x + ': ' + this.y + ' hours';

}

},

plotOptions : {

line : {

dataLabels : {

enabled : true

}

}

},

legend : {

layout : 'vertical',

align : 'right',

verticalAlign : 'top',

x : -10,

y : 100,

borderWidth : 0

},

series : [{

name : 'Hours Worked',

data : result.data,

showInLegend : false

}]

});

}

function draw\_chart() {

var data = get\_search\_param();

$.post(url.data, data, function(result) {

chart1(result);

});

}

function get\_checked\_month() {

if ($('.chkall').attr('checked') == 'checked')

return 0;

else {

var a = [];

$('.chkmonth:checked').each(function(idx, elm) {

a.push($(this).val());

});

return (a == [] ? 0 : a);

}

}

function get\_search\_param() {

var month = get\_checked\_month();

var param = {

year : $('#id\_year').val()

};

if (month == 0)

param['month'] = 0;

else

param['month[]'] = month;

return param;

}

function uncheck\_all\_month() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chkmonth').removeAttr('checked');

}

function uncheck\_all() {

var a = $(this).attr('checked');

if (a == 'checked')

$('.chkall').removeAttr('checked');

}

function init() {

$('#id\_gen').click(draw\_chart);

$('.chkall').click(uncheck\_all\_month);

$('.chkmonth').click(uncheck\_all);

utils.init\_alert\_dialog('#dialog-message');

utils.bind\_hover($('#id\_gen'));

draw\_chart();

}

function load() {

return menu.get('/user/workhours/chart/', init);

}

return {

load : load

};

}());

E:\workspace\payroll\_rails\app\assets\javascripts\tpl\label\_error.html

<div id="error\_<%= field %>" class="ui-state-error"><%= msg %></div>

E:\workspace\payroll\_rails\app\assets\javascripts\tpl\label\_error\_inline.html

<span id="error\_<%= field %>" class="ui-state-error" style="margin-left: 10px"><%= msg %></span>