

```
<?xml version="1.0" encoding="utf-8" ?>
<j:jelly trim="false" xmlns:j="jelly:core" xmlns:g="glide"
    xmlns:j2="null" xmlns:g2="null">

<link href="4ecef19db29c850c4ec38ff9d96196d.cssdbx" type="text/css" rel="stylesheet"/>

    <g2:evaluate var="jvar_not_important"
        expression="
            var show_kb_menu = gs.getProperty('glide.knowman.show_kb_menu', 'true');
            var roles = gs.getProperty('glide.knowman.show_kb_menu.roles');
            if (roles == null || roles == "")
                ;
            else {
                var hasRole = gs.hasRole(roles);
                if (hasRole == false)
                    show_kb_menu = false;
            }
        " />

    <j2:if test="`${show_kb_menu}`">
        <j2:if test="`${sysparm_media} != 'print'`">
            <g2:evaluate var="jvar_not_important" expression="
                var knowHelp = new SNC.KnowledgeHelper();
                var show_kb_rating_options =
knowHelp.getBooleanProperty(current,'glide.knowman.show_rating_options',true);
                var roles = gs.getProperty('glide.knowman.show_rating_options.roles');
                if (roles == null || roles == "")
                    ;
                else {
                    var hasRole = gs.hasRole(roles);
                    if (hasRole == false)
                        show_kb_rating_options = false;
                }
            " />
        </j2:if>
    </j2:if>
</j:jelly>
```

```

    }
  "/>
<g2:evaluate var="jvar_not_important" jelly="true" expression="
    var knowHelp = new SNC.KnowledgeHelper();
    var show_kb_star_rating = knowHelp.getBooleanProperty(current, 'glide.knowman.show_star_rating', true);
    var roles = gs.getProperty('glide.knowman.show_star_rating.roles');
    if (roles == null || roles == "")
        ;
    else {
        var hasRole = gs.hasRole(roles);
        if (hasRole == false)
            show_kb_star_rating = false;
    }
"/>
<j2:if test="`${show_kb_star_rating}`">
    <g2:evaluate var="jvar_num_ratings">
        var count = new
        GlideAggregate('kb_feedback');
        count.addQuery('article',
            `${current.sys_id}`);
        count.addQuery('rating','!=','');
        count.addAggregate('COUNT');
        count.query();
        var ratings = 0;
        if (count.next()) {
            ratings = count.getAggregate('COUNT');
        }
        ratings;
    </g2:evaluate>
</j2:if>
<g:evaluate var="jvar_failure_message" expression="gs.getMessage('Article rating not submitted')" />
<j2:if test="`${show_kb_rating_options}`">

```

```
<script>
```

```
function test(){
alert("sdsdd");
}
function KbStarRatingClass() {
    var currentId = '${current.sys_id}';
    var currentRating = 0;

    function ajaxPost(value) {
        if (currentId) {
            var ajax = new GlideAjax('KBAjaxSNC');
            ajax.addParam('sysparm_type', 'saveStarRating');
            ajax.addParam('sysparm_rating', value);
            ajax.addParam('sysparm_id', currentId);
            var query = $j('#sysparm_search');
            var ts_queryId = "${ts_queryId}";
            ajax.addParam('sysparm_search', (query.length) ? escape(query.val()) : '');
            ajax.addParam("ts_queryId", ts_queryId);
            ajax.getXMLAnswer(postRatingDone);
        }
    }
    function postRatingDone(answer){
        var data = answer.evalJSON(true);
        if (!data) {
            return kbViewArticle.showErrorMessage('${jvar_failure_message}');
        }
        if (!data.success) {
            return kbViewArticle.showErrorMessage(data.message);
        }
    }
}
```

```

    }
}
function setStarsRating(value) {
    if (!isNaN(value)) {
        currentRating = Math.round(value);
    }
    if (currentRating > 0) {
        var ratingMessage = getUIMessage([currentRating]);
        $('#articleStarRatingGroup').attr("title", ratingMessage);
        $('#articleStarRatingGroup').attr("data-original-title", ratingMessage);
        $('.kb-star-rating label').filter(function() {
            return $(this).attr('data-index') <= currentRating;
        }).addClass('kb-active-star');
        if(this.selfRated) {
            $('.kb-star-rating label.kb-active-star').addClass('kb-active-star-
self').removeClass('kb-active-star');
        }
    }
    else{
        var message = "$[gs.getMessage('This article has not been rated yet. Use up and down
arrow keys to select and submit the rating.')]";
        $('#articleStarRatingGroup').attr("title", message);
        $('#articleStarRatingGroup').attr("data-original-title", message);
    }
}
function saveRating(value) {
    ajaxPost(value);
    this.selfRated = true;
}
return {
    setStarsRating: setStarsRating,
    saveRating: saveRating
}

```

CSS

```
    };
}
$(function() {
    // As the hex values of the Heisenberg icons keep changing it's impossible to reliably set the "content"

    // attribute of the :hover:before pseudo class in the stylesheet in order to display a full star instead of an
    // empty one when hovering over with the mouse.
    // Instead we set the pseudo to fetch its "content" from a "data-content" attribute. (see .kb-star-rating)
    var ele = $('#placeholder-icon-star')[0];
    if(ele $('[AND] window.getComputedStyle) {
        var content;
        var contentStyle = getComputedStyle(ele,':before');
        if(!contentStyle) {
            //Happens for FF in popovers
            contentStyle = window.parent.getComputedStyle(ele,':before');
        }
        content = contentStyle.getPropertyValue('content');
        if(content $('[AND] content != 'none') {
            content = content.replace("/g, "");
            $('[.kb-star-rating .icon-star-empty]').attr('data-content', content);
        }else if(content == 'none'){
            //This happens for popovers in IE
            useIconStarImageClasses();
        }
    }
}
// create an instance
var kbArticleRating = new KbStarRatingClass();
kbArticleRating.setStarsRating($[current.rating]);
$('[.kb-star-rating label]').click(function (event) {
    var targetIndex = $(event.currentTarget).attr('data-index');
    kbArticleRating.setStarsRating(targetIndex);
    $('#rated_value').val(targetIndex);
});
```

```

        kbArticleRating.saveRating(targetIndex);
        updateCurrentStarRatingForAria(targetIndex);
    });
    $j('.kb-star-rating input').on("focus",function(e) {
        var id = e.target.id;
        id = parseInt(id[id.length-1]);
        $j('.kb-star-rating label').filter(function() {
            return $j(this).attr('data-index') ${AMP}lt;= id;
        }).addClass('kb-hover-star');
    });
    $j('.kb-star-rating input').on("blur",function(e) {
        $j('.kb-star-rating label').removeClass('kb-hover-star');
    });
    $j('.kb-star-rating').on("keydown",function(e) {
        var id = e.target.id;
        var int_id = parseInt(id[id.length-1]);
        if(e.keyCode == 40) {
            int_id %= 5;
            if(int_id == 0){
                $j(this).find('label').removeClass('kb-hover-star');
            }
            $j('.kb-star-rating label').filter(function() {
                return $j(this).attr('data-index') ${AMP}lt;= int_id+1;
            }).addClass('kb-hover-star');
        }
        else if(e.keyCode == 38) {
            $j('.kb-star-rating label').filter(function() {
                return $j(this).attr('data-index') ${AMP}gt;= int_id;
            }).removeClass('kb-hover-star');
            if(int_id == 1){
                $j(this).find('label').addClass('kb-hover-star');
            }
        }
    });

```

```

    }
    else if(e.keyCode == 13) {
        kbArticleRating.setStarsRating(int_id);
        $('#rated_value').val(int_id);
        kbArticleRating.saveRating(int_id);
        updateCurrentStarRatingForAria(int_id);
    }
    else if(e.keyCode == 9) {
        $(this).find('label').removeClass('kb-hover-star');
    }
});
$('.kb-star-rating').on("mouseenter focusin", function(){
    removeToolTipForRatingParent("remove");
    $('.kb-star-rating label.kb-active-star').removeClass('kb-active-star');
    $('.kb-star-rating label.kb-active-star-self').removeClass('kb-active-star-self');
});
$('.kb-star-rating').on("mouseleave focusout",function() {
    kbArticleRating.setStarsRating();
});
$('.kb-star-rating label.icon-star-empty').on("mouseenter focusin", function(){
    $(this).addClass('kb-hover-star').prevAll().addClass('kb-hover-star');
});
$('.kb-star-rating label.icon-star-empty').on("mouseleave focusout",function() {
    $(this).removeClass('kb-hover-star').prevAll().removeClass('kb-hover-star');
});
<j2:if test="$[jvar_isMSIE8 || jvar_isMSIE7]">
    useIconStarImageClasses();
</j2:if>
function useIconStarImageClasses(){
    $('.kb-star-rating').addClass('kb-star-image');
    $('.icon-star-empty').mouseover(function() {
        $(this).addClass('kb-hover-star').prevAll().addClass('kb-hover-star');
    });
}

```

```

        }).mouseout(function() {
            $(this).removeClass('kb-hover-star').prevAll().removeClass('kb-hover-star');
        });
    }
    updateCurrentStarRatingForAria($[current.rating]);
});

function getUIMessage(params) {
    var message = "$[gs.getMessage('This article has {0} star rating.Use up and down arrow keys to
select and submit the rating.')]";

    return message.replace(/\{(\d+)\}/g, function() {
        return params[arguments[1]];
    });
};

function removeToolTipForRatingParent(){
    $('#articleStarRatingGroup').removeAttr("title","");
    $('#articleStarRatingGroup').removeAttr("data-original-title","");
}

function updateCurrentStarRatingForAria(currentRating){
    if(currentRating)
        $('#articleStarRatingGroup').attr("aria-label",
getUIMessage([Math.round(currentRating)]));
    else $('#articleStarRatingGroup').attr("aria-label","$[gs.getMessage('This article has not been
rated yet.Use up and down arrow keys to select and submit the rating.')]");
    if($[gs.getPreference('glide.ui.accessibility', 'false') == 'true']){
        $('#articleStarRatingGroup').attr('tabindex', '0');
    }
}
}
</script>
<j2:if test="[$[show_kb_star_rating]]">
    <input id="rating_value" type="hidden" value="[$[current.rating]]"/>

```

```
<input id="num_ratings" type="hidden" value="$[jvar_num_ratings]"/>
<input id="rated_value" type="hidden"/>
<g:evaluate>
    var readOnly = kbViewInfo.hideFeedbackOptions?'read-only:':";
</g:evaluate>
<g:set_if var="jvar_accessibility" test="$[gs.getPreference('glide.ui.accessibility', 'false') == 'true']" true="accessibility-
enabled" false="accessibility-disabled" />
<div role="application" id="articleStarRatingGroup" class="$[jvar_accessibility]" style="display:inline-flex">
    <span class="rating kb-star-rating $[readOnly]">
        <p>Please choose a value to rate this article.</p>
        <input type="radio" class="rating-input" id="rating-input-1-1" name="rating-input-1" aria-
label="$[gs.getMessage('{0}-star rating', 1)]" title="$[gs.getMessage('{0}-star rating', 1)]"/>
        <label data-index="1" for="rating-input-1-1" class="rating-star icon-star-empty"
title="$[gs.getMessage('{0}-star rating', 1)]"/>
        <input type="radio" class="rating-input" id="rating-input-1-2" name="rating-input-1" aria-
label="$[gs.getMessage('{0}-star rating', 2)]" title="$[gs.getMessage('{0}-star rating', 2)]"/>
        <label data-index="2" for="rating-input-1-2" class="rating-star icon-star-empty"
title="$[gs.getMessage('{0}-star rating', 2)]"/>
        <input type="radio" class="rating-input" id="rating-input-1-3" name="rating-input-1" aria-
label="$[gs.getMessage('{0}-star rating', 3)]" title="$[gs.getMessage('{0}-star rating', 3)]"/>
        <label data-index="3" for="rating-input-1-3" class="rating-star icon-star-empty"
title="$[gs.getMessage('{0}-star rating', 3)]"/>
        <input type="radio" class="rating-input" id="rating-input-1-4" name="rating-input-1" aria-
label="$[gs.getMessage('{0}-star rating', 4)]" title="$[gs.getMessage('{0}-star rating', 4)]"/>
        <label data-index="4" for="rating-input-1-4" class="rating-star icon-star-empty"
title="$[gs.getMessage('{0}-star rating', 4)]"/>
        <input type="radio" class="rating-input" id="rating-input-1-5" name="rating-input-1" aria-
label="$[gs.getMessage('{0}-star rating', 5)]" title="$[gs.getMessage('{0}-star rating', 5)]"/>
        <label data-index="5" for="rating-input-1-5" class="rating-star icon-star-empty "
title="$[gs.getMessage('{0}-star rating', 5)]"/>
    </span>
```

```
<span id="placeholder-icon-star" class="icon-star" style="visibility:hidden; position: absolute;"></span>
```

```
</div>
```

```
</j2:if>
```

```
</j2:if>
```

```
</j2:if>
```

```
<g2:evaluate var="jvar_rating_options_hidden" jelly="true" object="true">
```

```
var isRatingOptionsHidden = !(show_kb_star_rating ${AMP}${AMP} show_kb_rating_options);
```

```
isRatingOptionsHidden;
```

```
</g2:evaluate>
```

```
<j2:if test="${jvar_rating_options_hidden}">
```

```
<script>
```

```
//if show_kb_rating_options or show_kb_star_rating is hidden,
```

```
//then remove the extra padding-left:20px of kb-article-view-count
```

```
$('.kb-article-view-count').css('padding','0');
```

```
</script>
```

```
</j2:if>
```

```
</j2:if>
```

```
</j:jelly>
```