

```
/**
Author: Blake Harms
Version: 2.7
See http://developer.mindtouch.com/DekiScript/FAQ/
How_do_I..._Build_an_Integrated_Bug_and_Issue_Tracker for a changelog.
*/

dekiapi();
jquery.ui("smoothness");
// Gather options from parent (form)
var options = String.eval(page.parent.properties.options.text);
var params = string.deserialize(options.params);

<html><head>
<script type="text/javascript" src="http://developer.mindtouch.com/@api/
deki/files/4634/=jquery.autocomplete.pack.js"></script>
<link rel="stylesheet" type="text/css"
href="http://developer.mindtouch.com/@api/deki/files/
4635/=jquery.autocomplete.css" />
<script type="text/javascript">
Deki.$(document).ready(function() { //when the document is ready...
    Deki.$('.datepicker').datepicker();
    Deki.$('.resizable').resizable();
    Deki.$('#ticketProperties').find('input,
select,textarea').change(function() {
        $self = Deki.$(this);
        var prop = 'urn:custom.mindtouch.com#' + $self.attr('name');
// url that retrieves the page properties
        Deki.Api.ReadPageProperty(null, prop, function(result) {
            if(result.etag) { // page property exists, write over it.
                Deki.Api.UpdatePageProperty(result.href, $self.val(),
result.etag, function() {
                    reportSuccess($self);
                }, function(result) {
                    alert('An error occurred trying to update the store
(status: ' + result.status + ' - ' + result.text + ')');
                });
            } else { // page property doesn't exist, create one.
                Deki.Api.CreatePageProperty(null, prop, $self.val(),
function() {
                    reportSuccess($self);
                }, function(result) {
                    alert('An error occurred trying to create the store
(status: ' + result.status + ' - ' + result.text + ')');
                });
            }
        }, function(result) {
            alert('An error occurred trying to read the store (status: '
+ result.status + ' - ' + result.text + ')');
```

```

    });
  });
});
"</script>
<script type="text/javascript">
var reportSuccess = function($self) {
  $self.addClass('value_updated');
  setTimeout('$self.removeClass(\'\'value_updated\')',1000);
}
"</script>
<style type="text/css">
#ticketProperties table {
border-spacing:0;border-collapse:collapse;  }
td.label {  background-color:#F0F0F0; font-weight:bold; width:150px;
}
.value_updated {  background-color:#FFFA7 !important;  }
"</style>
</head></html>
<div id="top-nav-bar">
  <a href=(page.parent.uri) class='to-table'> "Back to form" </a>
</div>
<div id="ticketProperties">
  <table cellspacing="0" cellpadding="3" border="1" width="70%"
style="table-layout: fixed;">
    <tbody>
      foreach(var param in params){
        var type = param.value.type;
        if(! type){
          if(param.value.data is list || param.value is
map)    let type = 'select';
          if(!param.value.data || param.value.data is
str)    let type = 'text';
        }
        if(param.value.show == 'table' || param.key=='title')
continue;  // skip if not supposed to be seen here.
        <tr>
          <td class="label"> param .value.label ??
String.toCamelCase(param .key) </td>
          <td>
            var property =
Map.values (map.select (page.properties,
"$key=="..String.replace (String.toLowerCase (param .key), " ", "_") .."'")) [0];
            if (param.value.editable != false) {
              switch (type) {
                case 'suggest':
                  if (param.value.data is list) {
                    <input type="text"
name=(String.toLowerCase (String.replace (param.key, ' ', '_')))
style=(param.value.style.form) value=(param.value.data) ctor=(

```

```

"$this.autocomplete("..JSON.emit(param.value.data)..");") />
    }
    break;
    case 'text':
        <input type="text"
name=(String.toLowerCase(String.replace(param.key, ' ', '_')))
style=(param.value.style.ticket) value=(property.text) />
        break;
    case 'textarea':
        var resizable = (typeof
param.value.resizable == typeof null || param.value.resizable) ?
'resizable' : '';
        <textarea class=(resizable)
name=(String.toLowerCase(String.replace(param.key, ' ', '_')))
style=(param.value.style.ticket)>property.text</textarea>
        break;
    case 'datepicker':
        <input type="text" class="datepicker"
name=(String.toLowerCase(String.replace(param.key, ' ', '_')))
style=(param.value.style.ticket) value=(property.text) />
        break;
    case 'hidden':
        property.text;
        break;
    case 'title':
        <input type="text" class="title"
name=(String.toLowerCase(String.replace(param.key, ' ', '_')))
style=(param.value.style.ticket) value=(param.value.data) />
        break;
    case 'status': // follow through.
    case 'select':
        if((param.value.data is map) ||
(param.value.data is list)){
            <select
name=(String.toLowerCase(String.replace(param.key, ' ', '_')))
style=(param.value.style.ticket)>
                <option
value=(property.text)>property.text</option>
                <option value=""> ""</option>
                foreach(var option in
param.value.data) {
                    <option value=(option)>
option </option>
                }
            </select>
        }
        else{
            <span
style=(param.value.style.ticket)>param.value.data</span>
        }

```

```
                break;
            default: // ignore.
                break;
        }
    }
    else {
        <span
style=(param.value.style.ticket)>property.text</span>
        }
    </td>
</tr>
}
</tbody>
</table>
</div>
```