**Cashbook**

**Scripts Applied**

1. **Source of Receipt (f12) 🡪 On Load (Load source of receipts from database)**

var $http=$scope.srv.$http;

// Following query will get data from the fields of Data Entry Form. F1, F2 are the fields name. ExecutionID

// is the ProcessID of Data Entry Form

// If data of single field is to be get then you can write it as title (text) and value to fill combo box

var query= "Select Distinct F1 as title, F1 as value from DATA\_FORM where ExecutionID='1966059874' order by F1 ASC;";

var Url ="/data/Query?query=" + encodeURIComponent(query);

$http({

method: "GET",

url: Url,

data: {},

headers: {

contentType: "application/json; charset=utf-8"

}

}).success(function(data, status, headers, config){

// Following code will bind data with fields. Variable of "data" (array type) will contain data of all the fields.

// fill combo box

f12.options=data;

$scope.refresh();

}).error(function(data, status, headers, config){

});

1. **Receipt Head of Accounts (f11) and Payment Head of Accounts (f14) 🡪 On Load (Load head of accounts from database)**

var $http=$scope.srv.$http;

// Following query will get data from the fields of Data Entry Form. F1, F2 are the fields name. ExecutionID

// is the ProcessID of Data Entry Form

// If data of single field is to be get then you can write it as title (text) and value to fill combo box

var query= "Select Distinct F1 as title, F1 as value from DATA\_FORM where ExecutionID='426796427' order by F1 ASC;";

var Url ="/data/Query?query=" + encodeURIComponent(query);

$http({

method: "GET",

url: Url,

data: {},

headers: {

contentType: "application/json; charset=utf-8"

}

}).success(function(data, status, headers, config){

// Following code will bind data with fields. Variable of "data" (array type) will contain data of all the fields.

// fill combo box

f14.options=data;

$scope.refresh();

}).error(function(data, status, headers, config){

});

1. **Date (f1) 🡪 On Load (Show current date)**

var today = new Date();

var dd = today.getDate();

var mm = today.getMonth()+1; //January is 0!

var yyyy = today.getFullYear();

if(dd<10)

{

dd='0'+dd;

}

if(mm<10)

{

mm='0'+mm;

}

f1.value = mm+'/'+dd+'/'+yyyy;

1. **Date (f1) 🡪 On Submit (Checks applied if data is not entered)**

// If there is no value entered in Cash or Bank fields then form should not be submitted

if (f7.value === "" && f5.value === "" && f8.value === "" && f6.value === "")

{

alert("Enter value in Receipts or Payments.");

$('input[id="f7"]').focus();

return false;

}

// If Receipt Cash or Receipt Bank fields has value then option should be selected from combo of "Source of Receipt"

// Field of "Receipt Head of Accounts" is not mandatory

if (f7.value !== "" || f5.value !== "")

{

if (f12.value === "")

{

alert("Select [Source of Receipt]");

$('select[id="f12"]').focus();

return false;

}

}

// If Payment Cash or Payment Bank fields has value then option should be selected from combo of "Payment Head of Accounts"

if (f8.value !== "" || f6.value !== "")

{

if (f14.value === "")

{

alert("Select [Payment Head of Accounts]");

$('select[id="f14"]').focus();

return false;

}

}

// If value in combo of "Source of Receipt" is selected then prompt for value in Receipt Cash or Receipt Bank fields

if (f12.value !== "")

{

if (f7.value === "" && f5.value === "")

{

alert("Enter value for [Source of Receipt]");

$('input[id="f7"]').focus();

return false;

}

}

// If value in combo of "Payment Head of Accounts" is selected then prompt for value in Payment Cash or Payment Bank fields

if (f14.value !== "")

{

if (f8.value === "" && f6.value === "")

{

alert("Enter value for [Payment Head of Accounts]");

$('input[id="f8"]').focus();

return false;

}

}

**OLD CODE (check of null is included)**

// If there is no value entered in Cash or Bank fields then form should not be submitted

if ((f7.value === null || f7.value === "") && (f5.value === null || f5.value === "") && (f8.value === null || f8.value === "") && (f6.value === null || f6.value === ""))

{

alert("Enter some value.");

$('input[id="f7"]').focus();

return false;

}

// If Receipt Cash or Receipt Bank fields has value then option should be selected from combo of "Source of Receipt"

if ((f7.value !== null && f7.value !== "") || (f5.value !== null && f5.value !== ""))

{

if (f12.value === null || f12.value === "")

{

alert("Select [Source of Receipt]");

$('select[id="f12"]').focus();

return false;

}

}

// If Payment Cash or Payment Bank fields has value then option should be selected from combo of "Head of Accounts"

if ((f8.value !== null && f8.value !== "") || (f6.value !== null && f6.value !== ""))

{

if (f14.value === null || f14.value === "")

{

alert("Select [Head of Account]");

$('select[id="f14"]').focus();

return false;

}

}

// If value in combo of "Source of Receipt" is selected then prompt for value in Receipt Cash or Receipt Bank fields

if (f12.value !== null && f12.value !== "")

{

if ((f7.value === null || f7.value === "") && (f5.value === null || f5.value === ""))

{

alert("Enter value for [Source of Receipt]");

$('input[id="f7"]').focus();

return false;

}

}

// If value in combo of "Head of Accounts" is selected then prompt for value in Payment Cash or Payment Bank fields

if (f14.value !== null && f14.value !== "")

{

if ((f8.value === null || f8.value === "") && (f6.value === null || f6.value === ""))

{

alert("Enter value for [Head of Accounts]");

$('input[id="f8"]').focus();

return false;

}

}