**PayLoads for tracker & aggregate data push**

**For Tracker**

//For Register and Enrollment//

var teienroll = {

"trackedEntity": teId,

"orgUnit": orgunitId,

"attributes": [ {

"attribute": attributeId,

"value": attributevalue

},

{

"attribute": attributeId,

"value": attributevalue

}],

"enrollments": [ {

"orgUnit": orgunitId,

"program": programId,

"status" : "ACTIVE",

"enrollmentDate": date

} ]

}

$.ajax({

async: false,

type: "POST",

dataType: "json",

contentType: "application/json",

url: '../api/trackedEntityInstances/',

data: JSON.stringify(teienroll),

success: function (response) {

//for v26 response//

tei=response.response.reference;

if(tei == undefined)

{

//for v27 response//

tei=response.response.importSummaries[0].reference;

}

},

error: function (response) {

}

});

//For only enrollment//

var enroll = {

"enrollments": [ {

"orgUnit": orgunitId,

"program": programId,

"status" : "ACTIVE",

"trackedEntityInstance" : trackEntityInstance,

"enrollmentDate": date

} ]

}

$.ajax({

async: false,

type: "POST",

dataType: "json",

contentType: "application/json",

url: '../api/enrollments/',

data: JSON.stringify(enroll),

success: function (response) {

},

error: function (response) {

}

});

//For new Event creation//

var event = {

trackedEntityInstance: tei,

orgUnit: orgunitId,

programStage : programStageId,

program : programId,

eventDate: date,

status : "ACTIVE",

storedBy : "admin",

coordinate: {

latitude: parseFloat(value),

longitude: parseFloat(value)

},

dataValues:[]

};

event.dataValues.push({

dataElement: dataElementId,

value: valueofdataelement

})

$.ajax({

async: false,

type: "POST",

dataType: "json",

contentType: "application/json",

url: '../api/events/',

data: JSON.stringify(event),

success: function (response) {

},

error: function (response) {

}

});

//To UPDATE existing Event//

var event = {

trackedEntityInstance: tei,

orgUnit: orgunitId,

programStage : programStageId,

program : programId,

eventDate: date,

status : "ACTIVE",

storedBy : "admin",

coordinate: {

latitude: parseFloat(value),

longitude: parseFloat(value)

},

dataValues:[]

};

event.dataValues.push({

dataElement: dataElementId,

value: valueofdataelement

})

$.ajax({

async: false,

type: "PUT",

dataType: "json",

contentType: "application/json",

url: '../api/events'+eventID,

data: JSON.stringify(event),

success: function (response) {

},

error: function (response) {

}

});

//To push values in blank attributes//

//Get all tei and loop through a particular tei attributes and push all attributes at once//

“../api/trackedEntityInstances.json?ou=ouID&program=prgID&programStartDate=2018-01-01&programEndDate=2018-04-30&skipPaging=true”

var tei\_value = {

attribute: "attibuteID",

created: "dateAndtime",

lastUpdated: "dateAndtime",

storedBy: "userName",

value: value,

valueType: "TEXT"

};

var getAllattributes= fromProgram.filter(function(obj) {

return attribute.indexOf(obj) == -1;

});

if(getAllattributes[t] == "attibuteID"){

allTei.attributes.push(tei\_value);

RegistrationService.registerOrUpdate(trackdata,optionSets, attributesById).then(function(response){

if (response.response.status == "SUCCESS"){

return;

}

});

}

//Another method To push values in blank attributes//

//Get all tei and loop through a particular tei attributes and push all attributes at once//

“../api/trackedEntityInstances.json?ou=ouID&program=prgID&programStartDate=2018-01-01&programEndDate=2018-04-30&skipPaging=true”

var teiAttributeValue = {

attribute: "attibuteID",

created: "dateAndtime",

displayName: "nameOfAttribute",

lastUpdated: "dateAndtime",

storedBy: "userName",

value: value,

valueType: "TEXT"

};

allTei.trackedEntityInstances[i].attributes.push(teiAttributeValue);

$.ajax({

async: false,

type: "PUT",

dataType: "json",

contentType: "application/json",

url: '../api/trackedEntityInstances/'+ onetei[i],

data: JSON.stringify(allTei.trackedEntityInstances[i]),

success: function (response) {

},

error: function (response) {

}

});

//For creating Data Element//

var de = {

aggregationType:"NONE",

domainType:"TRACKER",

name:newDataElement,

shortName: newDataElement,

code:newDataElement,

description:newDataElement,

formName:newDataElement,

valueType: "TEXT"

}

$.ajax({

async: false,

type: "POST",

dataType: "json",

contentType: "application/json",

url: '../api/dataElements/',

data: JSON.stringify(de),

success: function (response) {

console.log("Imported dataElement:",newDataElement);

},

error: function (response) {

console.log("Not Imported dataElement:",newDataElement);

}

});

//For creating Program Stage Section//

var psSection = {

name:psSectionName,

dataElement: {

"id": "OBlbusIfGB3"

},

sortOrder:0

}

$.ajax({

async: false,

type: "POST",

dataType: "json",

contentType: "application/json",

url: '../api/programStageSections/',

data: JSON.stringify(psSection),

success: function (response) {

responseId = response.response.uid;

console.log("Imported dataElement:",+ psSectionName);

},

error: function (response) {

console.log("Not Imported dataElement:", + psSectionName);

}

});

//For creating Program Stage//

var ps = {

name:psName,

description:psName,

minDaysFromStart:0,

program: {

"id": "p5lnyTwdbgK"

},

programStageDataElements: [{

dataElement: {

"id": "OBlbusIfGB3"

}

}],

programStageSections:[{

"id": responseId

}]

}

$.ajax({

async: false,

type: "POST",

dataType: "json",

contentType: "application/json",

url: '../api/programStages/',

data: JSON.stringify(ps),

success: function (response) {

console.log("Imported dataElement:",+ psName);

},

error: function (response) {

console.log("Not Imported dataElement:", + psName);

var text = "Not Imported dataElement:" + psName;

$('#result').append(text);

}

});

**For Aggregate**

//For pushing data into dataValueSet //

var dataValueSet = {

"dataSet": dataSetId,

"period": selectedPeriod,

"dataValues": []

};

dataValueSet.dataValues.push({

orgUnit: ouId,

categoryOptionCombo: categoryComboId,

period: selectedPeriod,

dataElement: deId,

value: value

})

$.ajax({

async: false,

type: 'post',

dataType: 'json',

contentType: "application/json",

url: '../api/dataValueSets',

data: JSON.stringify(dataValueSet),

success: function (response) {

},

warning: function (response) {

},

error: function (response) {

}

});

//For dataSet completion //

for (var z in allOu) {

var dataSetCompleteParams = {

"ds": dataSet,

"pe": selectedPeriod,

"ou": allOu[z],

"multiOu": false

};

$.ajax({

async: false,

type: 'post',

dataType: 'json',

url: '../../25/completeDataSetRegistrations',

data: dataSetCompleteParams,

success: function (response) {

},

error: function (response) {

}

});

}

//For pushing single dataValue //

var dataValue = {  
       'de' : de,  
       'co' : co,  
       'ou' : ou,  
       'pe' : pe,  
       'value' : value  
   };  
  
   var cc = dhis2.de.getCurrentCategoryCombo();  
   var cp = dhis2.de.getCurrentCategoryOptionsQueryValue();  
     
   if ( cc && cp )  
   {  
    dataValue.cc = cc;  
    dataValue.cp = cp;  
   }  
  
  $.ajax( {  
 url: '../api/dataValues',  
 data: dataValue,  
 type: 'post',  
 success: handleSuccess,  
 error: handleError  
 });  
  
  
  function handleSuccess()  
   {  
console.log( "SUCCESS");  
   }  
  
  function handleError( xhr, textStatus, errorThrown )  
   {  
    if ( 409 == xhr.status || 500 == xhr.status ) // Invalid value or locked  
    {  
    console.log( "ERROR");  
    }  
    else // Offline, keep local value  
    {  
           console.log( "ERROR");  
    }  
   }

// update single eventdatavalue and post evenetdatavalue

for PUT

**var** updateEventDataValue = {  
 **event**: eventUid,  
 **program**: programUid,  
 **trackedEntityInstance**: teiUid,  
 **dataValues**: [  
 {  
 **dataElement**: deUid,  
 **value**: deValue  
 }  
 ]  
};  
  
**console**.log( **"Event DataValue fro Update :"** , updateEventDataValue );  
  
**$**.ajax({  
 **async**:**false**,  
 **type**: **"PUT"**,  
 **url**: **'../../../api/events/'** +eventUid + **'/'** + deUid,  
 **dataType**: **"json"**,  
 **contentType**: **"application/json"**,  
 **data**: **JSON**.stringify(updateEventDataValue),  
 success: **function**(updateEventResponse){  
 **console**.log(**"Update Event with UID :"**, updateEventResponse + **"--"** + updateEventDataValue.**event** + **" -- "** + deUid + **" -- "** + deValue );  
 },  
 error: **function**(updateEventResponse){  
 **console**.log(**"Not Update Event with UID:"**, updateEventResponse + **"--"** + updateEventDataValue.**event** + **" -- "** + deUid + **" -- "** + deValue );  
 }  
});

for post

**var** eventDataValue = {  
 **event**: $scope.**data**.**rows**.**selectedArray**[j].eventuid,  
 **orgUnit**: $scope.**data**.**rows**.**selectedArray**[j].orguid,  
 **program**: $scope.**data**.**rows**.**selectedArray**[j].programuid,  
 **trackedEntityInstance**: $scope.**data**.**rows**.**selectedArray**[j].teiuid,  
 **dataValues**: [  
 {  
 **dataElement**: $scope.**batchNoDeUid**,  
 **value**: batchNo  
 },  
 {  
 **dataElement**: $scope.**dispatchDateDeUid**,  
 **value**: dispatchDate  
 },  
 {  
 **dataElement**: $scope.**dispatchStatusDeUid**,  
 **value**: dispatchStatus  
 },  
 {  
 **dataElement**: $scope.**receivedDateDeUid**,  
 **value**: receivedDate  
 }  
 ]  
};  
  
**if**( eventDataValue.**dataValues**.**length** != 0){  
 **$**.ajax({  
 **async**: **false**,  
 **type**: **"POST"**,  
 **dataType**: **"json"**,  
 **contentType**: **"application/json"**,  
 **url**: **'../../../api/events/'**,  
 **data**: **JSON**.stringify(eventDataValue),  
 success: **function** (response) {  
 **console**.log(**"Update Event with UID :"**,eventDataValue.**event**);  
 },  
 error: **function** (response) {  
 **console**.log(**"Not Update Event with UID:"**,eventDataValue.**event**);  
 }  
 });  
}