

```

var rc;
var tc;
var loops =0;
var nbOfToken = 0;
var handledTest = 0;
var contadorpunch =0;
var purchaseRequisitiondataArray = [];
var purchaseRequisitionData = [];
Process.PreventApproval();
var action = Data.GetActionName();
Log.Info("action:"+action);
Process.SetTimeOut(11000);

var cd_po = Variable.GetValueAsString("CD_PO");
var new_pp = Variable.GetValueAsString("NEW_PP");
var hashMapRequisitionType = {
    PG:"Pagamentos",
    CD:"Compras Delegadas",
    CE:"Compras Emergenciais",
    CAB:"Compras abaixo de R$4.000,00",
    NQ:"Compras acima de R$4.000,00",
    HQ:"Legado"
};

if(action == "GeraRelatorio")

{
    var arrayOfPromises = [];
    createPurchaseRequisitionData()
    .Then(function(successMsg){
        Log.Verbose(successMsg);
        for(var i = 0 ; i < purchaseRequisitionData.length ; i++) {
            var tempPurchaseRequisitionData = purchaseRequisitionData[i];
            switch(tempPurchaseRequisitionData.LoopType) {
                case 1:

arrayOfPromises.push(queryGestaoDeContratos(tempPurchaseRequisitionData,tempPurchaseRequisitionData.requisitionType,""));
                break;
                case 2:

arrayOfPromises.push(queryGestaoDeContratos(tempPurchaseRequisitionData,tempPurchaseRequisitionData.requisitionType,tempPurchaseRequisitionData.ruid));
                break;
                case 3:
                    arrayOfPromises.push(queryPO(tempPurchaseRequisitionData));
                break;
                case 4:
                    arrayOfPromises.push(queryTabPR(tempPurchaseRequisitionData));
                break;
                default:
                    //Log.Info("default");
            }
        }
    });
}

```

```

        break;
    }
}
Log.Verbose("arrayOfPromises.length:" + arrayOfPromises.length);

Sys.Helpers.Promise.All(arrayOfPromises)
.Then(function(success){
    Log.Verbose("build my email");
    Log.Info("Size purchaseRequisitiondataArray: " +
purchaseRequisitiondataArray.length);
    BuildMyEmail(purchaseRequisitiondataArray);

}).Catch(function(err){
    Log.Verbose("Email not build");
    Log.Verbose(err);
});

})
.Catch(Log.Error);
//.Then(function(outputPurchaseRequisitiondataArray) {

// })
//all my rejects will trigger the catch and return a string, I want to log it directly thanks to the
Log.Error method
//.Catch(Log.Error)
}

/**
 * Create query filter based on data on the form
 * @param query query on the PR process
 */
function buildPRQueryFilter(query) {
    var dtinicio = Data.GetValue("Inicio__");
    var dtfim = Data.GetValue("Fim__");

    var mesini= zeroFill(dtinicio.getMonth()+1,2);
    var mesfim= zeroFill(dtfim.getMonth()+1,2);
    var diaini= zeroFill(dtinicio.getDate(),2);
    var diafim= zeroFill(dtfim.getDate(),2);

    var ini = dtinicio.getFullYear() + "-" + mesini + "-" + diaini + " 00:00:00";
    var fim = dtfim.getFullYear() + "-" + mesfim + "-" + diafim + " 23:59:59";
    Log.Error(ini);
    Log.Error(fim);

    var dep;

    var filterdep;
    if(Data.GetValue("Departamento__") == "Todos" || !Data.GetValue("Departamento__"))
    {
        filterdep = "(Departamento__=*)";
    }
}

```

```

//dep = "";
/*
    (Departamento__=Administrativo) (Departamento__=Call Center)
(Departamento__=Cobranca)(Departamento__=Comercial)      (Departamento__=Compliance)
    (Departamento__=Compras)(Departamento__=Contabilidade) (Departamento__=Contas
a Pagar)      (Departamento__=Controladoria) (Departamento__=Credito)
(Departamento__=Diretoria) (Departamento__=Fiscal) (Departamento__=GPTW)
(Departamento__=Juridico) (Departamento__=Juridico Corporativo)(Departamento__=Marketing)
    (Departamento__=Operacoes)(Departamento__=Presidencia)(Departamento__=Produtos)
    (Departamento__=Recursos Humanos)(Departamento__=Regional Brasilia)
    (Departamento__=Regional Porto Alegre) (Departamento__=Regional Recife)
(Departamento__=Regional Rio de Janeiro)(Departamento__=Regional Sao Paulo)
    (Departamento__=Remarketing)(Departamento__=Risco) (Departamento__=Seguros)
(Departamento__=Sistemas)(Departamento__=Tesouraria)
*/
//filterdep = "(!("Departamento__=Administrativo) (Departamento__=Call Center)
(Departamento__=Cobranca)(Departamento__=Comercial)      (Departamento__=Compliance)
    (Departamento__=Compras)(Departamento__=Contabilidade) (Departamento__=Contas
a Pagar)      (Departamento__=Controladoria) (Departamento__=Credito)
(Departamento__=Diretoria) (Departamento__=Fiscal) (Departamento__=GPTW)
(Departamento__=Juridico) (Departamento__=Juridico Corporativo)(Departamento__=Marketing)
    (Departamento__=Operacoes)(Departamento__=Presidencia)(Departamento__=Produtos)
    (Departamento__=Recursos Humanos)(Departamento__=Regional Brasilia)
    (Departamento__=Regional Porto Alegre) (Departamento__=Regional Recife)
(Departamento__=Regional Rio de Janeiro)(Departamento__=Regional Sao Paulo)
    (Departamento__=Remarketing)(Departamento__=Risco) (Departamento__=Seguros)
(Departamento__=Sistemas)(Departamento__=Tesouraria))";
}
else
{
    dep = Data.GetValue("Departamento__");
    filterdep = "(Departamento__="+dep+")";
}
Log.Info(filterdep);
//var filterdep = "(Departamento__="+dep+")";
//var filterini = "(OriginalSubmitDateTime>="+ini+")";
var filterini = "(SubmitDateTime>="+ini+")";
//var filterfim = "(OriginalSubmitDateTime<="+fim+")";
var filterfim = "(SubmitDateTime<="+fim+")";

//Filter data
//Log.Info("Inicio: " + filter);
query.AddFilter("&",filterdep);
query.AddFilter("&",filterini);
query.AddFilter("&",filterfim);
    query.AddFilter("&", "(Deleted=0)");
//query.AddFilter("&", "RequisitionNumber__>"+new_pp);
//Query.AddFilter("&", "(RecipientType=USF)")
if(Data.GetValue("Requisicao_de_Compras__"))
{
    query.AddFilter("&",
"(RequisitionNumber__="+Data.GetValue("Requisicao_de_Compras__")+")");
}

```

```

    //Log.Info("(RequisitionNumber__="+Data.GetValue("Requisicao_de_Compras__")+")");
}

if(Data.GetValue("Fornecedor__"))
{
    query.AddFilter("&", "(VendorNumber__="+Data.GetValue("Fornecedor__")+")");
}

}

/**
 * return the first object of the purchaseRequisitionDataArray
 * @returns
 */
function initpurchaseRequisitionDataArray() {
    return {
        'TipoDocumento': "Tipo Documento",
        'TipoCompra': "Tipo de Compra",
        'SubtipoCompra': "Subtipo de Compra",
        'RC': "Número da RC",
        'STATE': "Status",
        'Requisitante': "Requisitante da RC",
        'ResponsavelRC': "Responsável pela RC",
        'DataRC': "Data de abertura da RC",
        'DescItem' : "Descrição do item",
        'CondPgto' : "Condições de Pagamento",
        'StatusRC': "Status da RC",
        'Aprovador': "Nome do Aprovador",
        'PO': "Número do PO",
        'DataPO': "Data do PO",
        'Buyer': "Usuario que gerou o PO",
        'Departamento': "Departamento",
        'Montante': "Valor da RC/PO",
        'Saldo': "Saldo da RC/PO",
        'NomeFornecedor': "Nome do Fornecedor",
        'CNPJFornecedor': "CNPJ do Fornecedor",
        'Contrato': "Número do Contrato",
        'ContratoStatus': "Status do Contrato",
        'ContratoOwner' : "Responsavel",
        'DataInicio': "Data de Inicio do Contrato",
        'DataFim': "Data de Fim do Contrato",
        'DataNF': "Data da NF",
        'DataPG': "Data do Pagamento",
        'PassoNF': "Passo da NF",
        'MontanteNF': "Montante da NF",
        'StatusNF:' : "Status da NF"
    };
}

function initpurchaseRequisitionData() {
    return {

```

```

        'TipoDocumento': '',
        'TipoCompra': '',
        'SubtipoCompra': '',
        'RC': '',
        'STATE': '',
        'Requisitante': '',
        'ResponsavelRC': '',
        'DataRC': '',
        'DescItem' : '',
        'CondPgto' : '',
        'StatusRC': '',
        'Aprovador': '',
        'PO': '',
        'DataPO': '',
        'Buyer': '',
        'Departamento': '',
        'Montante': '',
        'Saldo': '',
        'NomeFornecedor': '',
        'CNPJFornecedor': '',
        'Contrato': '',
        'ContratoStatus': '',
        'ContratoOwner' : '',
        'DataInicio': '',
        'DataFim': '',
        'DataNF': '',
        'DataPG': '',
        'PassoNF': '',
        'MontanteNF': '',
        'StatusNF:' : '',
            'LoopType:' : 0,
            'Gravado:' : 0,
            'requisitionType' : ''
    };
}

```

```

/**

```

```

* Here i'm creating your array of data, named purchaseRequisitiondataArray to ease the clarity of the code.

```

```

* it will be populated with purchaseRequisitionData, which is the result of my query on the process Purchase requisition

```

```

* inside this query, I will trigger another Promise "queryGestaoDeContratos". The result of the Promise "queryGestaoDeContratos" will resolve the "createPurchaseRequisitionData"

```

```

* @returns

```

```

*/

```

```

function createPurchaseRequisitionData() {
    return Sys.Helpers.Promise.Create(function (resolve, reject) {
        //init synchronizer
        var nbOfRecords = 0;

```

```

        //inicializa OBJ

```

```

//End inicializa OBJ

var transport ; // in this case purchaseRequisitionData = data[x] and
purchaseRequisitionDataArray = data
var tc;
    purchaseRequisitionDataArray.push(initpurchaseRequisitionDataArray());
var queryM = Process.CreateQueryAsProcessAdmin();
queryM.Reset();
queryM.SetOptionEx("Limit=5000");
//Factorise the PR Query filter to gain visibility
buildPRQueryFilter(queryM);

queryM.SetSpecificTable("CDNAME#Purchase requisition");
//queryM.SetSpecificTable("CDNAME#Purchase requisition Audit");
queryM.SetAttributesList("*");
//queryM.AddAttribute("DataFile");
queryM.SetSearchInArchive(true);
    queryM.SetSortOrder("RequisitionNumber__ ASC");
transport = queryM.MoveFirst();
    var tipocompra;
if(transport) {
    nbOfRecords = queryM.GetRecordCount();
    Log.Error("nbOfRecords:"+nbOfRecords);
    if(nbOfRecords == 0)
    {
        Data.SetValue("Mensagem__","Não foram encontradas RCs com os
dados informados");
    }
    transport = queryM.MoveNext();
    //The loop is performed here, no need to create a loop over the getRecordCount() method
    while(transport) {
        purchaseRequisitionData.push(initpurchaseRequisitionData());

        var vars = transport.GetUninheritedVars();
        //Log.Verbose("Loop: " + loops + " RC: " +
vars.GetValue_String("RequisitionNumber__",0));

        //var vars = transport.GetUninheritedVars();
        if(vars.GetValue_String("RequisitionNumber__",0))
        {
            rc = vars.GetValue_String("RequisitionNumber__",0);
            Log.Verbose(rc);
        }

        //purchaseRequisitionDataArray.push({'RC':vars.GetValue_String("RequisitionNumber__",0)});

```

/* On the part above, I missed the logic, RequisitionNumber__ is requested three times and it's bad for performances

/* Maybe you should store it in variable and manipulate this variable ?

```
purchaseRequisitionData[loops].TipoDocumento = "Compras";
    if(vars.GetValue_String("RequisitionNumber__",0))
    {
purchaseRequisitionData[loops].RC = vars.GetValue_String("RequisitionNumber__",0);
    }
    var requisitionType;
    if(vars.GetValue_String("RequisitionType__",0))
    {
requisitionType = vars.GetValue_String("RequisitionType__",0);
    }

    if(vars.GetValue_String("STATE",0))
    {

        var s = vars.GetValue_String("STATE",0);

        if(s == "0")
        {
            purchaseRequisitionData[loops].STATE = "Processando";

        }
        if(s == "10")
        {
            purchaseRequisitionData[loops].STATE = "Processando"

        }
        if(s == "30")
        {
            purchaseRequisitionData[loops].STATE = "Processando";

        }
        if(s == "40")
        {
            purchaseRequisitionData[loops].STATE = "Processando";

        }
        if(s == "50")
        {
            purchaseRequisitionData[loops].STATE = "Processando";

        }
        if(s == "60")
        {
            purchaseRequisitionData[loops].STATE = "Processando";

        }
        if(s == "70")
        {
```

```

switch(requisitionType)
{

    case "PG":

switch(vars.GetValue_String("RequisitionStatus__",0))
{
    case
    "New":purchaseRequisitionData[loops].STATE = "Nova";
        break;
    case "To Approve":
        break;
    case "Approved":
        break;
}
break;

    case "CD":

switch(vars.GetValue_String("RequisitionStatus__",0))
{
    case "New":
        break;
    case "To Approve":
        break;
    case "Approved":
        break;
}
break;

    case "CE":

switch(vars.GetValue_String("RequisitionStatus__",0))
{
    case "New":
        break;
    case "To Approve":
        break;
    case "Approved":
        break;
}
break;

    case "CAB":

switch(vars.GetValue_String("RequisitionStatus__",0))

```



```

        {
            case "New":
                purchaseRequisitionData[loops].STATE = "Nova";
                break;
            case "To Approve":
                purchaseRequisitionData[loops].STATE = "Para aprovar";
                break;
            case "Approved":
                purchaseRequisitionData[loops].STATE = "Para receber";
                break;
        }
        break;
        case "NQ":
            switch(vars.GetValue_String("RequisitionStatus__",0))
            {
                case "New":
                    purchaseRequisitionData[loops].STATE = "Nova";
                    break;
                case "To Approve":
                    purchaseRequisitionData[loops].STATE = "Para aprovar";
                    break;
                case "Approved":
                    purchaseRequisitionData[loops].STATE = "Para gerar PO";
                    break;
                case "Provide
                    quotation": purchaseRequisitionData[loops].STATE = "Para Cotar";
                    break;
                case "To verify":
                    purchaseRequisitionData[loops].STATE = "Para verificar";
                    break;
            }
            default:
                break;
        }

        //purchaseRequisitionData[loops].STATE = "Para validar";
    }
    if(s == "80")
    {
        purchaseRequisitionData[loops].STATE = "Ocupado";
    }
    if(s == "90")
    {
        purchaseRequisitionData[loops].STATE = "Processando";
    }
    if(s == "100")
    {

```

```

        purchaseRequisitionData[loops].STATE = "Sucesso";

    }
    if(s == "200")
    {
        purchaseRequisitionData[loops].STATE = "Falha";

    }
    if(s == "300")
    {
        purchaseRequisitionData[loops].STATE = "Cancelado";

    }
    if(s == "400")
    {
        purchaseRequisitionData[loops].STATE = "Rejeitado";

    }
}

```

```

        if(requisitionType)
        {
            tipocompra = requisitionType;
            tc = tipocompra;
        }
        else
        {
            tipocompra = "";
            tc = tipocompra;
        }
    }
}

```

//Using a hashmap instead of switch in order to gain performances, just a proposition, feel free to use it.

```

        if(requisitionType)
        {
            purchaseRequisitionData[loops].TipoCompra =
            hashMapRequisitionType[requisitionType];
            purchaseRequisitionData[loops].requisitionType = requisitionType;

        }
        else
        {
            purchaseRequisitionData[loops].TipoCompra = "";
        }
    }
}

```

```

//Continue creating your purchaseRequisitionData
        if(vars.GetValue_String("Tipo_Compra__",0) && (requisitionType
        == "CD" || requisitionType == "CE"))
        {
            purchaseRequisitionData[loops].SubtipoCompra =
            vars.GetValue_String("Tipo_Compra__",0);
        }
    }
}

```

```

else
{
purchaseRequisitionData[loops].SubtipoCompra = "";
}
if(vars.GetValue_String("SubmitDateTime",0))
{
//purchaseRequisitionData[loops].DataRC =
vars.GetValue_String("OriginalSubmitDateTime",0);
var dtsb = new Date(vars.GetValue_Date("SubmitDateTime",0));

purchaseRequisitionData[loops].DataRC =
(adicionaZero(dtsb.getDate().toString()) + "/" + (adicionaZero(dtsb.getMonth()+1).toString()) + "/"
+ dtsb.getFullYear() + " " + dtsb.getHours() + ":" + dtsb.getMinutes());

}
else
{
purchaseRequisitionData[loops].DataRC = "";
}
//Log.Info("RS: " + vars.GetValue_String("RequisitionStatus__",0));
if(vars.GetValue_String("RequisitionStatus__",0))
{

purchaseRequisitionData[loops].StatusRC =
vars.GetValue_String("RequisitionStatus__",0);
}
else
{
purchaseRequisitionData[loops].StatusRC = "";
}
if(vars.GetValue_String("Aprovador__",0))
{
purchaseRequisitionData[loops].Aprovador = vars.GetValue_String("Aprovador__",0);
}
else
{
purchaseRequisitionData[loops].Aprovador = "";
}
//This Contrato is needed to test and create your second query
if(vars.GetValue_String("Codigo_Contrato__",0))
{
purchaseRequisitionData[loops].Contrato =
vars.GetValue_String("Codigo_Contrato__",0);
}
else
{
purchaseRequisitionData[loops].Contrato = "";
}
if(vars.GetValue_String("DataIni__",0))
{
purchaseRequisitionData[loops].DataInicio = vars.GetValue_String("DataIni__",0);
}

```

```

else
{
purchaseRequisitionData[loops].DataInicio = "";
}
if(vars.GetValue_String("DataFim___",0))
{
purchaseRequisitionData[loops].DataFim = vars.GetValue_String("DataFim___",0);
}
else
{
purchaseRequisitionData[loops].DataFim = "";
}
if(vars.GetValue_String("VendorName___",0))
{
purchaseRequisitionData[loops].NomeFornecedor =
vars.GetValue_String("VendorName___",0);
}
else
{
purchaseRequisitionData[loops].NomeFornecedor = "";
}
if(vars.GetValue_String("VendorVatNumber___",0))
{
purchaseRequisitionData[loops].CNPJFornecedor =
vars.GetValue_String("VendorVatNumber___",0);
}
else
{
purchaseRequisitionData[loops].CNPJFornecedor = "";
}
if(vars.GetValue_String("Departamento___",0))
{
purchaseRequisitionData[loops].Departamento =
vars.GetValue_String("Departamento___",0);
}
else
{
purchaseRequisitionData[loops].Departamento = "";
}
if(vars.GetValue_String("TotalNetAmount___",0))
{
purchaseRequisitionData[loops].Montante =
vars.GetValue_String("TotalNetAmount___",0);
}
else
{
purchaseRequisitionData[loops].Montante = 0;
}

```

// [...] keep creating your purchaseRequisitionData

```

        if(vars.GetValue_String("RequesterName__",0)) //RequesterName__
        {
            purchaseRequisitionData[loops].Requisitante =
vars.GetValue_String("RequesterName__",0);
        }
        else
        {
            purchaseRequisitionData[loops].Requisitante = "";
        }
        if(vars.GetValue_String("CItemsDescricao__",0)) //CItemsDescricao__
        {
            purchaseRequisitionData[loops].DescItem =
vars.GetValue_String("CItemsDescricao__",0);
        }
        else
        {
            purchaseRequisitionData[loops].DescItem = "";
        }
        if(vars.GetValue_String("CondicoesPagamento__",0))
//CondicoesPagamento__
        {
            purchaseRequisitionData[loops].CondPgto =
vars.GetValue_String("CondicoesPagamento__",0);
        }
        else
        {
            purchaseRequisitionData[loops].CondPgto = "";
        }

        purchaseRequisitionData[loops].ResponsavelRC =
vars.GetValue_String("OwnerID",0);
        var c = purchaseRequisitionData[loops].ResponsavelRC.split(",");
        var owner = c[0].substring(3,c[0].length);
        purchaseRequisitionData[loops].ResponsavelRC = owner;

        //Additional logic : I add the the property "link" and "State" in order to store it and test
        later in my loop line 28
        if(vars.GetValue_String("Link_Contrato_Persistente__",0))
        {
            purchaseRequisitionData[loops].link =
vars.GetValue_String("Link_Contrato_Persistente__",0);
        }
        else
        {
            purchaseRequisitionData[loops].link = "";
        }
        if(vars.GetValue_String("State",0))
        {
            purchaseRequisitionData[loops].State =
vars.GetValue_String("State",0);

```

```

    }
    else
    {
        purchaseRequisitionData[loops].State = "";
    }

    purchaseRequisitionData[loops].ContratoStatus = "";
    purchaseRequisitionData[loops].ContratoOwner = "";

    purchaseRequisitionData[loops].LoopType = 0;
    purchaseRequisitionData[loops].Gravado = 0;
    //purchaseRequisitionDataArray.push(purchaseRequisitionData);
    if(purchaseRequisitionData[loops].Contrato) {
        //Log.Error("Com contrato " +
purchaseRequisitionData[loops].RC);
        //H3
        purchaseRequisitionData[loops].LoopType = 1;
        // synchronizer.Register(function(token) {

            //     for(var w = 0; w < loops;w++){

                //     if(purchaseRequisitionData[w].LoopType == 1 &&
purchaseRequisitionData[w].Gravado == 0)
                //     {
                //         purchaseRequisitionData[w].Gravado = 1;
                //     queryGestaoDeContratos(purchaseRequisitionData[w],tipocompra,"")
                //         .Then(function(successMsg) {
                //             Log.Verbose(successMsg + " RC:" +
purchaseRequisitionData[w].RC);
                //         })
                //         .Catch(function(err){
                //     Log.Error("Erro GC 1 " + err);

                //     });

                //     }
                //     token.Use();
                //     }

            // });

            // .Then(function(result) {
            //     purchaseRequisitionDataArray.push(result)
            //     resolve(result);
            // })
            // .Catch(Log.Error);

        } else if(vars.GetValue_String("Link_Contrato_Persistente__",0))

```

```

        {
            //Log.Error("Link contrato " +
purchaseRequisitionData[loops].RC);
            //H3

            var link =
vars.GetValue_String("Link_Contrato_Persistente__",0);

            var ruid = link.substring(link.length - 18,link.length);
purchaseRequisitionData[loops].ruid = ruid;
purchaseRequisitionData[loops].LoopType = 2;

            // synchronizer.Register(function(token) {
            //     for(var w = 0; w < loops;w++){

                //     if(purchaseRequisitionData[w].LoopType == 2 &&
purchaseRequisitionData[w].Gravado == 0)
                //     {
                //         purchaseRequisitionData[w].Gravado = 1;

                //
queryGestaoDeContratos(purchaseRequisitionData[w],tipocompra,purchaseRequisitionData[loops].
ruid)
                //         .Then(function(successMsg) {
                //             Log.Verbose(successMsg + " RC:" +
purchaseRequisitionData[w].RC);
                //         })
                //         .Catch(function(err){
                //             Log.Error("Erro GC " + err);

                //     });

                //     }
                //     token.Use();
                // }

            // });

            //}.Then(function(result) {
            //purchaseRequisitionDataArray.push(result)
            //resolve(purchaseRequisitionDataArray);
            //})
            //}.Catch(Log.Error);

        }
    else {

        //Log.Error("Sem contrato ");

```

```

        if(((tc == "NQ") &&
(purchaseRequisitionData[loops].StatusRC == "Completed"))
        || ((tc == "CD") &&
(purchaseRequisitionData[loops].StatusRC ==
"Completed"))&&(purchaseRequisitionData[loops].RC > cd_po))
        || ((tc == "HQ") &&
(purchaseRequisitionData[loops].StatusRC == "Completed"))) )
        {

                purchaseRequisitionData[loops].LoopType = 3;
                //Log.Error("CompletaPO");
                // synchronizer.Register(function(token) {
                //         for(var w = 0; w < loops;w++){

                // if(purchaseRequisitionData[w].LoopType == 3 &&
purchaseRequisitionData[w].Gravado == 0)
                // {
                // purchaseRequisitionData[w].Gravado = 1;
                // queryPO(purchaseRequisitionData[w])
                // .Then(function(successMsg) {
                //         Log.Verbose(successMsg + " RC:" +
purchaseRequisitionData[w].RC);

                // })
                // .Catch(function(err){
                //         Log.Error("Erro queryPO " + err);

                // });

                // }
                // token.Use();
                // }

                // });

                //}.Then(function(resultOfAnotherQuery) {
                //Reassign the result of another query in the
purchaseRequisitionData object
                //purchaseRequisitionData =
resultOfAnotherQuery;
                //resolve(purchaseRequisitionData)
                //})
                //}.Catch(Log.Error)

        }
        else if(((tc == "PG") &&
((purchaseRequisitionData[loops].StatusRC=="Approved"))||
(purchaseRequisitionData[loops].StatusRC=="Completed")))
                ||(tc == "CD" &&
(purchaseRequisitionData[loops].StatusRC=="Approved" ||
purchaseRequisitionData[loops].StatusRC=="Completed" ||

```



```

purchaseRequisitionData[loops].StatusRC=="Waiting for deliver") &&
purchaseRequisitionData[loops].RC <= cd_po)
    ||(tc == "CE" &&
(purchaseRequisitionData[loops].StatusRC=="Approved" ||
purchaseRequisitionData[loops].StatusRC=="Completed" ||
purchaseRequisitionData[loops].StatusRC=="Waiting for deliver"))
    ||(tc == "CAB" &&
(purchaseRequisitionData[loops].StatusRC=="Approved" ||
purchaseRequisitionData[loops].StatusRC=="Completed" ||
purchaseRequisitionData[loops].StatusRC=="Waiting for deliver")))
    {
        //Log.Error("Completa PR");

        //H3

        purchaseRequisitionData[loops].LoopType = 4;
        // synchronizer.Register(function(token) {
        //     for(var w = 0; w < loops;w++){

                //     if(purchaseRequisitionData[w].LoopType == 4
&& purchaseRequisitionData[w].Gravado == 0)
                //     {
                //         purchaseRequisitionData[w].Gravado = 1;
                //         queryTabPR(purchaseRequisitionData[w])
                //             .Then(function(successMsg) {
                //                 Log.Verbose(successMsg + " RC:" +
purchaseRequisitionData[w].RC);

                //     })
                //     .Catch(function(err){
                //         Log.Error("Erro tabpr " + err);

                //     });

                // }
                // token.Use();
                // }

                // });

                //}.Then(function(resultOfAnotherAltQuery) {
                //    //Reassign the result of another query in the
purchaseRequisitionData object
                //purchaseRequisitionData =
                //resolve(purchaseRequisitionData)
                //})
                //}.Catch(Log.Error)
            }
            else {

```

```

//Log.Error("GravaRC " + tc + " " +
purchaseRequisitionData[loops].StatusRC);
/////Teste
/*
var po = "";
var dtpo = "";//var dtpo =
purchaseRequisitionData.DataPO; //Error
var by = "";//var by =
purchaseRequisitionData.Buyer; //Error
var saldo = "";//var saldo =
purchaseRequisitionData.Saldo; //Error

var dtnf = "";
var dtpg = "";
var st = "";
var mon = "";

var tc = purchaseRequisitionData.TipoCompra;
var stc = purchaseRequisitionData.SubtipoCompra;
var rc = purchaseRequisitionData.RC
var dtrc = purchaseRequisitionData.DataRC;
var src = purchaseRequisitionData.StatusRC;
var ap = purchaseRequisitionData.Aprovador;
var dept = purchaseRequisitionData.Departamento;
var monc = purchaseRequisitionData.Montante;
var nfor = purchaseRequisitionData.NomeFornecedor;
var cnpj = purchaseRequisitionData.CNPJFornecedor;
var req = purchaseRequisitionData.Requisitante; //Error
var desc = purchaseRequisitionData.DescItem; //Error
var rrc = purchaseRequisitionData.ResponsavelRC;

//Error

var condp = purchaseRequisitionData.CondPgto;

//Error

var con = purchaseRequisitionData.Contrato;
var dtini = purchaseRequisitionData.DataInicio;
var dtfim = purchaseRequisitionData.DataFim;
var cons = "";
var cono = "";
*/
//PO data
var po;
if(purchaseRequisitionData[loops].PO)
{po = purchaseRequisitionData[loops].PO;}
else
{po = "";}

var dtpo;
if(purchaseRequisitionData[loops].DataPO)
{dtpo = purchaseRequisitionData[loops].DataPO; }

```

```
else  
{dtpo="";}
```

```
var by ;  
if(purchaseRequisitionData[loops].Buyer)  
{by = purchaseRequisitionData[loops].Buyer;}  
else  
{by="";}
```

```
var saldo;  
if(purchaseRequisitionData[loops].Saldo)  
{saldo = purchaseRequisitionData[loops].Saldo; }  
else  
{saldo = 0;}
```

```
//NF data
```

```
var dtmf;  
if(purchaseRequisitionData[loops].DataNF)  
{dtmf = purchaseRequisitionData[loops].DataNF;}  
else  
{dtmf="";}
```

```
var dtpg;  
if(purchaseRequisitionData[loops].DataPG)  
{dtpg = purchaseRequisitionData[loops].DataPG;}  
else  
{dtpg="";}
```

```
var st;  
if(purchaseRequisitionData[loops].PassoNF)  
{st = purchaseRequisitionData[loops].PassoNF;}  
else  
{st="";}
```

```
var mon;  
if(purchaseRequisitionData[loops].MontanteNF)  
{mon = purchaseRequisitionData[loops].MontanteNF;}  
else  
{mon=0;}
```

```
//PR data
```

```
//var tc;  
if(purchaseRequisitionData[loops].TipoCompra)  
{tc = purchaseRequisitionData[loops].TipoCompra; }  
else  
{tc="";}
```

```
var stc;  
if(purchaseRequisitionData[loops].SubtipoCompra)
```

```

        {stc =
purchaseRequisitionData[loops].SubtipoCompra;}
        else
        {stc = "";}

        var rc;
        if(purchaseRequisitionData[loops].RC)
        {rc = purchaseRequisitionData[loops].RC;}
        else
        {rc="";}

        var dtrc;
        if(purchaseRequisitionData[loops].DataRC)
        {dtrc = purchaseRequisitionData[loops].DataRC;}
        else
        {dtrc="";}

        var src;
        if(purchaseRequisitionData[loops].StatusRC)
        {src =purchaseRequisitionData[loops].StatusRC;}
        else
        {src="";}

        var ap;
        if(purchaseRequisitionData[loops].Aprovador)
        {ap = purchaseRequisitionData[loops].Aprovador;}
        else
        {ap="";}

        var dept;
        if(purchaseRequisitionData[loops].Departamento)
        {dept =
purchaseRequisitionData[loops].Departamento;}
        else
        {dept="";}

        var monc;
        if(purchaseRequisitionData[loops].Montante)
        {monc = purchaseRequisitionData[loops].Montante;}
        else
        {monc = 0;}

        var nfor;
        if(purchaseRequisitionData[loops].NomeFornecedor)
        {nfor =
purchaseRequisitionData[loops].NomeFornecedor;}
        else
        {nfor = "";}

        var cnpj;
        if(purchaseRequisitionData[loops].CNPJFornecedor)

```

```

        {cnpj =
purchaseRequisitionData[loops].CNPJFornecedor;}
        else
        {cnpj = "";}

        var req;
        if(purchaseRequisitionData[loops].Requisitante)
        {req = purchaseRequisitionData[loops].Requisitante;}
        else
        {req = "";}

        var desc;
        if(purchaseRequisitionData[loops].DescItem)
        {desc = purchaseRequisitionData[loops].DescItem;}
        else
        {desc = "";}

        var rrc;
        if(purchaseRequisitionData[loops].ResponsavelRC)
        {rrc =
purchaseRequisitionData[loops].ResponsavelRC;}
        else
        {rrc="";}

        var condp;
        if(purchaseRequisitionData[loops].CondPgto)
        {condp = purchaseRequisitionData[loops].CondPgto;}
        else
        {condp = "";}

        var con;
        if(purchaseRequisitionData[loops].Contrato)
        {con = purchaseRequisitionData[loops].Contrato;}
        else
        {con = "";}

        var dtini;
        if(purchaseRequisitionData[loops].DataInicio)
        {dtini = purchaseRequisitionData[loops].DataInicio;}
        else
        {dtini = "";}

        var dtfim;
        if(purchaseRequisitionData[loops].DataFim)
        {dtfim = purchaseRequisitionData[loops].DataFim;}
        else
        {dtfim="";}

        //Contrato data
        var cons;
        if(purchaseRequisitionData[loops].ContratoStatus)

```

```

        { cons =
purchaseRequisitionData[loops].ContratoStatus;}
        else
        { cons="";}

        var cono;
        if(purchaseRequisitionData[loops].ContratoOwner)
        { cono =
purchaseRequisitionData[loops].ContratoOwner;}
        else
        { cono="";}

        var state;
        if(purchaseRequisitionData[loops].STATE)
        { state = purchaseRequisitionData[loops].STATE;}
        else
        { state="";}

        //synchronizer.Register(function(token) {
        purchaseRequisitionDataArray.push({'STATE': state,
'TipoDocumento':"Compras", 'PO':po, 'DataNF':dtnf,
'DataPG':dtpg,'StatusNF':st,'MontanteNF':mon,'TipoCompra':tc,'SubtipoCompra':stc,'RC':rc,'DataR
C':dtrc,'StatusRC':src,'Aprovador':ap,'DataPO':dtpo,'Buyer':by,'Departamento':dept,'Montante':monc
,'Saldo':saldo,'NomeFornecedor':nfor,'CNPJFornecedor':cnpj,'Contrato':con,'DataInicio':dtini,'DataF
im':dtfim,'ContratoStatus':cons,'ContratoOwner':cono,'Requisitante':req,'DescItem':desc,'Responsav
elRC':rrc, 'CondPgto':condp});
        // token.Use();
        //});

    }

}

```

//As we have the while condition, the loop will be created over all the results of the query on the process Purchase Requisition

```

        transport = queryM.MoveNext();
        loops++;
    }
    if(loops == nbOfRecords) {
        resolve("Youhou");
    }
    else{
        reject("Diferenca loops:" + loops + " nbOfRecords: "+nbOfRecords);
    }
    Log.Error("Loops" + loops);

```

```
Log.Error("Size purchaseRequisitionData: " +  
purchaseRequisitionData.length);
```

```
//Return the array of the result from the first query in order to send it to the next query for  
edition.
```

```
} else {  
    Log.Error("Nao fez query de RCs");
```

```
    if (transport.GetLastError())  
    {  
        reject(transport.GetLastErrorMessages());  
    }  
}
```

```
});  
}
```

```
function queryGestaoDeContratos(purchaseRequisitionData,tipocompra,ruid) {  
    return Sys.Helpers.Promise.Create(function (resolve, reject) {
```

```
        var tablec = "CDNAME#Gestao de Contratos";
```

```
        var filterc;
```

```
        if(purchaseRequisitionData.Contrato)  
        {
```

```
            filterc = "(Numero_Documento__="+purchaseRequisitionData.Contrato+"";
```

```
        }  
        else  
        {
```

```
            filterc = "(RuidEx=CD#GGBW2F6F9A3."+ruid+""; //PROD
```

```
            //filterc = "(RuidEx=CD#PHQG1F8FBFD."+ruid+"";
```

```
        }
```

```
        var QueryC = Process.CreateQueryAsProcessAdmin();
```

```
        QueryC.Reset();
```

```
        QueryC.SetSpecificTable(tablec);
```

```
        QueryC.SetFilter(filterc);
```

```
        QueryC.SetAttributesList("");
```

```
        QueryC.SetSearchInArchive(true);
```

```
        if (QueryC.MoveFirst())
```

```
        {
```

```
            //Data.SetValue("Retorno__",QueryC.GetRecordCount());
```

```
            var transportc = QueryC.MoveNext();
```

```

        if (transportc) {
            //Log.Verbose("Contrato: " + purchaseRequisitionData.Contrato);
            var varsc = transportc.GetUninheritedVars();
            //purchaseRequisitionData.ContratoStatus = varsc.GetValue_String("Status__",0);

            purchaseRequisitionData.ContratoOwner =
varsc.GetValue_String("OwnerID",0);
            var c = purchaseRequisitionData.ContratoOwner.split(",");
            var owner = c[0].substring(3,c[0].length);
            purchaseRequisitionData.ContratoOwner = owner;

            if(varsc.GetValue_String("Status__",0))
            {
                purchaseRequisitionData.ContratoStatus =
varsc.GetValue_String("Status__",0);
            }
            else
            {
                purchaseRequisitionData.ContratoStatus = "";
            }

            //purchaseRequisitionDataArray.push(purchaseRequisitionData);
            //Log.Verbose("Tipo Compra: " +
purchaseRequisitionData.TipoCompra);
            //This piece of code has been updated cause we stored this information in the Purchase
            Requisition object thanks to the first query
            //Log.Error("GC " + tipocompra);
            if(((tipocompra == "NQ") &&
(purchaseRequisitionData.StatusRC=="Completed"))
                ||((tipocompra == "CD") &&
(purchaseRequisitionData.StatusRC=="Completed")&&(purchaseRequisitionData.RC >cd_po))
                ||((tipocompra == "HQ") &&
(purchaseRequisitionData.StatusRC=="Completed")) )
            {
                //Log.Error("GC PO NQ " +
purchaseRequisitionData.StatusRC);

                queryPO(purchaseRequisitionData)
                .Then(function(successMsg) {
                    //Log.Verbose("Success Query PO");
                    resolve("Query GC successful");
                })
                .Catch(function(err){
                    //Log.Verbose("Error Query PO");
                    resolve("Erro QueryPO "+ err);
                })

                //Then(function(resultOfAnotherQuery) {

```



```

//Reassign the result of another query in the
purchaseRequisitionData object
resultOfAnotherQuery;

//purchaseRequisitionData =
//resolve(purchaseRequisitionData)
//})
//.Catch(Log.Error)
}
else if(((tipocompra == "PG" &&
purchaseRequisitionData.StatusRC=="Approved")||(tipocompra == "PG" &&
purchaseRequisitionData.StatusRC=="Completed"))
||(tipocompra == "CD" &&
(purchaseRequisitionData.StatusRC=="Approved" ||
purchaseRequisitionData.StatusRC=="Completed" || purchaseRequisitionData.StatusRC=="Waiting
for deliver") && purchaseRequisitionData.RC <= cd_po)
||(tipocompra == "CE" &&
(purchaseRequisitionData.StatusRC=="Approved" ||
purchaseRequisitionData.StatusRC=="Completed" || purchaseRequisitionData.StatusRC=="Waiting
for deliver")))
||(tipocompra == "CAB" &&
(purchaseRequisitionData.StatusRC=="Approved" ||
purchaseRequisitionData.StatusRC=="Completed" || purchaseRequisitionData.StatusRC=="Waiting
for deliver")))
)
{
//Log.Error("GC PO PG " +
purchaseRequisitionData.StatusRC);

queryTabPR(purchaseRequisitionData)
.Then(function(successMsg) {
//Log.Verbose(successMsg);
//Log.Verbose("Success Query tab PR")
resolve("Query GC successful");
})
.Catch(function(err){
//Log.Verbose("Error Query tab PR")
resolve("Erro QueryTabPR " + err);
})

//.Then(function(resultOfAnotherAltQuery) {
//Reassign the result of another query in the
purchaseRequisitionData object
resultOfAnotherAltQuery;

//purchaseRequisitionData =
//resolve(purchaseRequisitionData)
//})
//.Catch(Log.Error)
}
else {
//Log.Error("GC PO Running " +
purchaseRequisitionData.StatusRC);

/*
var po = "";

```

```

purchaseRequisitionData.DataPO; //Error
purchaseRequisitionData.Buyer; //Error

purchaseRequisitionData.Saldo; //Error

var dtpo = ""; //var dtpo =
var by = ""; //var by =

var saldo = 0; //var saldo =

var dtmf = "";
var dtpg = "";
var st = "";
var mon = "";

var tc = purchaseRequisitionData.TipoCompra;
var stc = purchaseRequisitionData.SubtipoCompra;
var rc = purchaseRequisitionData.RC
var dtrc = purchaseRequisitionData.DataRC;
var src = purchaseRequisitionData.StatusRC;
var ap = purchaseRequisitionData.Aprovador;
var dept = purchaseRequisitionData.Departamento;
var monc = purchaseRequisitionData.Montante;
var nfor = purchaseRequisitionData.NomeFornecedor;
var cnpj = purchaseRequisitionData.CNPJFornecedor;
var req = purchaseRequisitionData.Requisitante; //Error
var desc = purchaseRequisitionData.DescItem; //Error
var rrc = purchaseRequisitionData.ResponsavelRC;

//Error

//Error

var condp = purchaseRequisitionData.CondPgto;

var con = purchaseRequisitionData.Contrato;
var dtini = purchaseRequisitionData.DataInicio;
var dtfim = purchaseRequisitionData.DataFim;
var cons = purchaseRequisitionData.ContratoStatus;
var cono = purchaseRequisitionData.ContratoOwner;
*/

//PO data
var po;
if(purchaseRequisitionData.PO)
{po = purchaseRequisitionData.PO;}
else
{po = "";}

var dtpo;
if(purchaseRequisitionData.DataPO)
{dtpo = purchaseRequisitionData.DataPO; }
else
{dtpo="";}

```

```

var by ;
if(purchaseRequisitionData.Buyer)
{by = purchaseRequisitionData.Buyer;}
else
{by="";}

var saldo;
if(purchaseRequisitionData.Saldo)
{saldo = purchaseRequisitionData.Saldo; }
else
{saldo = 0;}

//NF data

var dtnf;
if(purchaseRequisitionData.DataNF)
{dtnf = purchaseRequisitionData.DataNF;}
else
{dtnf="";}

var dtpg;
if(purchaseRequisitionData.DataPG)
{dtpg = purchaseRequisitionData.DataPG;}
else
{dtpg="";}

var st;
if(purchaseRequisitionData.PassoNF)
{st = purchaseRequisitionData.PassoNF;}
else
{st="";}

var mon;
if(purchaseRequisitionData.MontanteNF)
{mon = purchaseRequisitionData.MontanteNF;}
else
{mon=0;}

//PR data
var tc;
if(purchaseRequisitionData.TipoCompra)
{tc = purchaseRequisitionData.TipoCompra; }
else
{tc="";}

var stc;
if(purchaseRequisitionData.SubtipoCompra)
{stc = purchaseRequisitionData.SubtipoCompra;}
else
{stc = "";}

```

```
var rc;
if(purchaseRequisitionData.RC)
{rc = purchaseRequisitionData.RC;}
else
{rc="";}

var dtrc;
if(purchaseRequisitionData.DataRC)
{dtrc = purchaseRequisitionData.DataRC;}
else
{dtrc="";}

var src;
if(purchaseRequisitionData.StatusRC)
{src =purchaseRequisitionData.StatusRC;}
else
{src="";}

var ap;
if(purchaseRequisitionData.Aprovador)
{ap = purchaseRequisitionData.Aprovador;}
else
{ap="";}

var dept;
if(purchaseRequisitionData.Departamento)
{dept = purchaseRequisitionData.Departamento;}
else
{dept="";}

var monc;
if(purchaseRequisitionData.Montante)
{monc = purchaseRequisitionData.Montante;}
else
{monc = 0;}

var nfor;
if(purchaseRequisitionData.NomeFornecedor)
{nfor = purchaseRequisitionData.NomeFornecedor;}
else
{nfor = "";}

var cnpj;
if(purchaseRequisitionData.CNPJFornecedor)
{cnpj = purchaseRequisitionData.CNPJFornecedor;}
else
{cnpj = "";}

var req;
if(purchaseRequisitionData.Requisitante)
{req = purchaseRequisitionData.Requisitante;}
else
```

```

{req = "";}

var desc;
if(purchaseRequisitionData.DescItem)
{desc = purchaseRequisitionData.DescItem;}
else
{desc = "";}

var rrc;
if(purchaseRequisitionData.ResponsavelRC)
{rrc = purchaseRequisitionData.ResponsavelRC;}
else
{rrc="";}

var condp;
if(purchaseRequisitionData.CondPgto)
{condp = purchaseRequisitionData.CondPgto;}
else
{condp = "";}

var con;
if(purchaseRequisitionData.Contrato)
{con = purchaseRequisitionData.Contrato;}
else
{con = "";}

var dtini;
if(purchaseRequisitionData.DataInicio)
{dtini = purchaseRequisitionData.DataInicio;}
else
{dtini = "";}

var dtfim;
if(purchaseRequisitionData.DataFim)
{dtfim = purchaseRequisitionData.DataFim;}
else
{dtfim="";}

//Contrato data
var cons;
if(purchaseRequisitionData.ContratoStatus)
{cons = purchaseRequisitionData.ContratoStatus;}
else
{cons="";}

var cono;
if(purchaseRequisitionData.ContratoOwner)
{cono = purchaseRequisitionData.ContratoOwner;}
else
{cono="";}

var state;

```

```

if(purchaseRequisitionData.STATE)
{state = purchaseRequisitionData.STATE;}
else
{state="";}

```

```

//purchaseRequisitionDataArray.push({"TipoDocumento":"Compras", 'PO':po, 'DataNF':dtnf,
'DataPG':dtpg,'StatusNF':st,'MontanteNF':mon,'TipoCompra':tc,'SubtipoCompra':stc,'RC':rc,'DataR
C':dtrc,'StatusRC':src,'Aprovador':ap,'DataPO':dtpo,'Buyer':by,'Departamento':dept,'Montante':monc
,'Saldo':saldo,'NomeFornecedor':nfor,'CNPJFornecedor':cnpj,'Contrato':con,'DataInicio':dtini,'DataF
im':dtfim,'ContratoStatus':cons,'ContratoOwner':cono,'Requisitante':req,'DescItem':desc,'Responsav
elRC':rrc, 'CondPgto':condp});

```

```

purchaseRequisitionDataArray.push({'STATE' : state,
'TipoDocumento':"Compras",
'PO':po,'TipoCompra':tc,'SubtipoCompra':stc,'RC':rc,'DataRC':dtrc,'StatusRC':src,'Aprovador':ap,'D
ataPO':dtpo,'Buyer':by,'Departamento':dept,'Montante':monc,'Saldo':saldo,'NomeFornecedor':nfor,'C
NPJFornecedor':cnpj,'Contrato':con,'DataInicio':dtini,'DataFim':dtfim,'ContratoStatus':cons,'Contrat
oOwner':cono,'Requisitante':req,'DescItem':desc,'ResponsavelRC':rrc, 'CondPgto':condp});

```

```

//Log.Verbose("success queryGestaoDeContratos");
resolve("success queryGestaoDeContratos");
}

```

```

} else {

```

```

//PO data

```

```

var po;
if(purchaseRequisitionData.PO)
{po = purchaseRequisitionData.PO;}
else
{po = "";}

```

```

var dtpo;
if(purchaseRequisitionData.DataPO)
{dtpo = purchaseRequisitionData.DataPO; }
else
{dtpo="";}

```

```

var by ;
if(purchaseRequisitionData.Buyer)
{by = purchaseRequisitionData.Buyer;}
else
{by="";}

```

```

var saldo;
if(purchaseRequisitionData.Saldo)
{saldo = purchaseRequisitionData.Saldo; }
else
{saldo = 0;}

```

```
//NF data
```

```
var dtnf;  
if(purchaseRequisitionData.DataNF)  
{dtnf = purchaseRequisitionData.DataNF;}  
else  
{dtnf="";}
```

```
var dtpg;  
if(purchaseRequisitionData.DataPG)  
{dtpg = purchaseRequisitionData.DataPG;}  
else  
{dtpg="";}
```

```
var st;  
if(purchaseRequisitionData.PassoNF)  
{st = purchaseRequisitionData.PassoNF;}  
else  
{st="";}
```

```
var mon;  
if(purchaseRequisitionData.MontanteNF)  
{mon = purchaseRequisitionData.MontanteNF;}  
else  
{mon=0;}
```

```
//PR data
```

```
var tc;  
if(purchaseRequisitionData.TipoCompra)  
{tc = purchaseRequisitionData.TipoCompra; }  
else  
{tc="";}
```

```
var stc;  
if(purchaseRequisitionData.SubtipoCompra)  
{stc = purchaseRequisitionData.SubtipoCompra;}  
else  
{stc = "";} 
```

```
var rc;  
if(purchaseRequisitionData.RC)  
{rc = purchaseRequisitionData.RC;}  
else  
{rc="";}
```

```
var dtrc;  
if(purchaseRequisitionData.DataRC)  
{dtrc = purchaseRequisitionData.DataRC;}  
else  
{dtrc="";}
```

```
var src;
```

```
if(purchaseRequisitionData.StatusRC)
{src =purchaseRequisitionData.StatusRC;}
else
{src="";}

var ap;
if(purchaseRequisitionData.Aprovador)
{ap = purchaseRequisitionData.Aprovador;}
else
{ap="";}

var dept;
if(purchaseRequisitionData.Departamento)
{dept = purchaseRequisitionData.Departamento;}
else
{dept="";}

var monc;
if(purchaseRequisitionData.Montante)
{monc = purchaseRequisitionData.Montante;}
else
{monc = 0;}

var nfor;
if(purchaseRequisitionData.NomeFornecedor)
{nfor = purchaseRequisitionData.NomeFornecedor;}
else
{nfor = "";}

var cnpj;
if(purchaseRequisitionData.CNPJFornecedor)
{cnpj = purchaseRequisitionData.CNPJFornecedor;}
else
{cnpj = "";}

var req;
if(purchaseRequisitionData.Requisitante)
{req = purchaseRequisitionData.Requisitante;}
else
{req = "";}

var desc;
if(purchaseRequisitionData.DescItem)
{desc = purchaseRequisitionData.DescItem;}
else
{desc = "";}

var rrc;
if(purchaseRequisitionData.ResponsavelRC)
{rrc = purchaseRequisitionData.ResponsavelRC;}
else
{rrc="";}
```



```
var condp;  
if(purchaseRequisitionData.CondPgto)  
{condp = purchaseRequisitionData.CondPgto;}  
else  
{condp = "";}  

```

```
var con;  
if(purchaseRequisitionData.Contrato)  
{con = purchaseRequisitionData.Contrato;}  
else  
{con = "";}  

```

```
var dtini;  
if(purchaseRequisitionData.DataInicio)  
{dtini = purchaseRequisitionData.DataInicio;}  
else  
{dtini = "";}  

```

```
var dtfim;  
if(purchaseRequisitionData.DataFim)  
{dtfim = purchaseRequisitionData.DataFim;}  
else  
{dtfim="";}  

```

```
//Contrato data  
var cons;  
if(purchaseRequisitionData.ContratoStatus)  
{cons = purchaseRequisitionData.ContratoStatus;}  
else  
{cons="";}  

```

```
var cono;  
if(purchaseRequisitionData.ContratoOwner)  
{cono = purchaseRequisitionData.ContratoOwner;}  
else  
{cono="";}  

```

```
var state;  
if(purchaseRequisitionData.STATE)  
{state = purchaseRequisitionData.STATE;}  
else  
{state="";}  

```

```
//purchaseRequisitionDataArray.push({'TipoDocumento':"Compras", 'PO':po, 'DataNF':dtnf,  
'DataPG':dtpg,'StatusNF':st,'MontanteNF':mon,'TipoCompra':tc,'SubtipoCompra':stc,'RC':rc,'DataR  
C':dtrc,'StatusRC':src,'Aprovador':ap,'DataPO':dtpo,'Buyer':by,'Departamento':dept,'Montante':monc  
, 'Saldo':saldo,'NomeFornecedor':nfor,'CNPJFornecedor':cnpj,'Contrato':con,'DataInicio':dtini,'DataF  
im':dtfim,'ContratoStatus':cons,'ContratoOwner':cono,'Requisitante':req,'DescItem':desc,'Responsav  
elRC':rrc, 'CondPgto':condp});
```

```

        purchaseRequisitiondataArray.push({'STATE': state,
'TipoDocumento':"Compras",
'PO':po,'TipoCompra':tc,'SubtipoCompra':stc,'RC':rc,'DataRC':dtrc,'StatusRC':src,'Aprovador':ap,'D
ataPO':dtpo,'Buyer':by,'Departamento':dept,'Montante':monc,'Saldo':saldo,'NomeFornecedor':nfor,'C
NPJFornecedor':cnpj,'Contrato':con,'DataInicio':dtini,'DataFim':dtfim,'ContratoStatus':cons,'Contrat
oOwner':cono,'Requisitante':req,'DescItem':desc,'ResponsavelRC':rrc, 'CondPgto':condp});
        //Log.Verbose("success queryGestaoDeContratos");
        //resolve("success queryGestaoDeContratos");

        resolve("Nao achou Contrato " + filterc);
    }
    } else {
        reject("Move First GC error");
    }
});
}

```

```

function queryPO(purchaseRequisitionData) {
    return Sys.Helpers.Promise.Create(function (resolve, reject) {

        //Log.Error("QueryPO RC:" + purchaseRequisitionData.RC);
        var tablePO = "CDNAME#Purchase order";
        var filterPO = "(RequisitionNumber__="+purchaseRequisitionData.RC+)";
        var QueryPO = Process.CreateQueryAsProcessAdmin();
        QueryPO.Reset();
        QueryPO.SetSpecificTable(tablePO);
        QueryPO.SetFilter(filterPO);
        QueryPO.SetAttributesList("*");
        QueryPO.SetSearchInArchive(true);
        if (QueryPO.MoveFirst())
        {

            //Data.SetValue("Retorno__",QueryC.GetRecordCount());
            var transportPO = QueryPO.MoveNext();
            if (transportPO) {

                var varspo = transportPO.GetUninheritedVars();
                //Log.Verbose("PO " + varspo.GetValue_String("OrderNumber__",0));
                if(varspo.GetValue_String("OrderNumber__",0))
                {
                    purchaseRequisitionData.PO =
varspo.GetValue_String("OrderNumber__",0);
                }
                else
                {
                    purchaseRequisitionData.PO = "";
                }
                if(varspo.GetValue_String("Data_Pedido__",0))
                {

```

```

        purchaseRequisitionData.DataPO =
varspo.GetValue_String("Data_Pedido___",0);
    }
    else
    {
        purchaseRequisitionData.DataPO = "";
    }
    if(varspo.GetValue_String("BuyerName___",0))
    {
        purchaseRequisitionData.Buyer =
varspo.GetValue_String("BuyerName___",0);
    }
    else
    {
        purchaseRequisitionData.Buyer = "";
    }

```

//This piece of code has been updated cause we stored this information in the Purchase Requisition object thanks to the first query

```

        queryTabPO(purchaseRequisitionData)
        .Then(function(successMsg) {
            //Log.Verbose(successMsg);
            resolve("Query PO successful");
        }).Catch(function(err){
            reject(err)
        })
        //Then(function(resultOfAnotherQuery) {
            //Reassign the result of another query in the purchaseRequisitionData object
            //purchaseRequisitionData = resultOfAnotherQuery;
            //resolve(purchaseRequisitionData)
        //})
        //Catch(Log.Error)

    } else {
        //Log.Verbose("Não achou PO");
        reject("queryPO error");
    }
} else {
    //Log.Verbose("PO movefirst error");
    reject("Move First queryPO error");
}

});
}

```

```

function queryTabPO(purchaseRequisitionData) {

```

```

return Sys.Helpers.Promise.Create(function (resolve, reject) {

    var tabletpo = "AP - Purchase order - Headers__";
    var filtertpo = "(OrderNumber__="+purchaseRequisitionData.PO+"")";
    var QueryTPO = Process.CreateQueryAsProcessAdmin();
    QueryTPO.Reset();
    QueryTPO.SetSpecificTable(tabletpo);
    QueryTPO.SetFilter(filtertpo);
    QueryTPO.SetAttributesList("*");
    QueryTPO.SetSearchInArchive(true);
    if (QueryTPO.MoveFirst())
    {
        //Data.SetValue("Retorno__",QueryC.GetRecordCount());
        var transporttpo = QueryTPO.MoveNextRecord();
        if (transporttpo) {
            //Log.Verbose("TabPO ");
            var varstbpo = transporttpo.GetVars();

            var oa = 0.0;
            var ia = 0.0;

            if(varstbpo.GetValue_Double("OrderedAmount__",0))
            {

                oa = varstbpo.GetValue_Double("OrderedAmount__",0);
            }

            if(varstbpo.GetValue_Double("InvoicedAmount__",0))
            {

                ia = varstbpo.GetValue_Double("InvoicedAmount__",0);
            }

            purchaseRequisitionData.Saldo = (oa - ia);
            purchaseRequisitionData.Saldo =
purchaseRequisitionData.Saldo.toFixed(2);
            //purchaseRequisitionData.Saldo = 999;
            /*
            //purchaseRequisitionDataArray.push(purchaseRequisitionData);
            var po = purchaseRequisitionData.PO;
            var dtpo = purchaseRequisitionData.DataPO; //Error
            var by = purchaseRequisitionData.Buyer; //Error

            var saldo = purchaseRequisitionData.Saldo; //Error

            var dtnf = "";
            var dtpg = "";
            var st = "";
            var mon = "";

            var tc = purchaseRequisitionData.TipoCompra;
            var stc = purchaseRequisitionData.SubtipoCompra;

```

```
var rc = purchaseRequisitionData.RC
var dtrc = purchaseRequisitionData.DataRC;
var src = purchaseRequisitionData.StatusRC;
var ap = purchaseRequisitionData.Aprovador;
var dept = purchaseRequisitionData.Departamento;
var monc = purchaseRequisitionData.Montante;
var nfor = purchaseRequisitionData.NomeFornecedor;
var cnpj = purchaseRequisitionData.CNPJFornecedor;
var req = purchaseRequisitionData.Requisitante; //Error
var desc = purchaseRequisitionData.DescItem; //Error
var rrc = purchaseRequisitionData.ResponsavelRC; //Error
var condp = purchaseRequisitionData.CondPgto; //Error
```

```
var con = purchaseRequisitionData.Contrato;
var dtini = purchaseRequisitionData.DataInicio;
var dtfim = purchaseRequisitionData.DataFim;
var cons = purchaseRequisitionData.ContratoStatus;
var cono = purchaseRequisitionData.ContratoOwner;
```

```
*/
```

```
//PO data
```

```
var po;
if(purchaseRequisitionData.PO)
{po = purchaseRequisitionData.PO;}
else
{po = "";}

```

```
var dtpo;
if(purchaseRequisitionData.DataPO)
{dtpo = purchaseRequisitionData.DataPO; }
else
{dtpo="";}

```

```
var by ;
if(purchaseRequisitionData.Buyer)
{by = purchaseRequisitionData.Buyer;}
else
{by="";}

```

```
var saldo;
if(purchaseRequisitionData.Saldo)
{saldo = purchaseRequisitionData.Saldo; }
else
{saldo = 0;}

```

```
//NF data
```

```
var dtnf;
```

```
if(purchaseRequisitionData.DataNF)
{dtnf = purchaseRequisitionData.DataNF;}
else
{dtnf="";}

var dtpg;
if(purchaseRequisitionData.DataPG)
{dtpg = purchaseRequisitionData.DataPG;}
else
{dtpg="";}

var st;
if(purchaseRequisitionData.PassoNF)
{st = purchaseRequisitionData.PassoNF;}
else
{st="";}

var mon;
if(purchaseRequisitionData.MontanteNF)
{mon = purchaseRequisitionData.MontanteNF;}
else
{mon=0;}

//PR data
var tc;
if(purchaseRequisitionData.TipoCompra)
{tc = purchaseRequisitionData.TipoCompra; }
else
{tc="";}

var stc;
if(purchaseRequisitionData.SubtipoCompra)
{stc = purchaseRequisitionData.SubtipoCompra;}
else
{stc = "";}

var rc;
if(purchaseRequisitionData.RC)
{rc = purchaseRequisitionData.RC;}
else
{rc="";}

var dtrc;
if(purchaseRequisitionData.DataRC)
{dtrc = purchaseRequisitionData.DataRC;}
else
{dtrc="";}

var src;
if(purchaseRequisitionData.StatusRC)
{src =purchaseRequisitionData.StatusRC;}
else
```

```
{src="";}
```

```
var ap;  
if(purchaseRequisitionData.Aprovador)  
{ap = purchaseRequisitionData.Aprovador;}  
else  
{ap="";}
```

```
var dept;  
if(purchaseRequisitionData.Departamento)  
{dept = purchaseRequisitionData.Departamento;}  
else  
{dept="";}
```

```
var monc;  
if(purchaseRequisitionData.Montante)  
{monc = purchaseRequisitionData.Montante;}  
else  
{monc = 0;}
```

```
var nfor;  
if(purchaseRequisitionData.NomeFornecedor)  
{nfor = purchaseRequisitionData.NomeFornecedor;}  
else  
{nfor = ""};}
```

```
var cnpj;  
if(purchaseRequisitionData.CNPJFornecedor)  
{cnpj = purchaseRequisitionData.CNPJFornecedor;}  
else  
{cnpj = ""};}
```

```
var req;  
if(purchaseRequisitionData.Requisitante)  
{req = purchaseRequisitionData.Requisitante;}  
else  
{req = ""};}
```

```
var desc;  
if(purchaseRequisitionData.DescItem)  
{desc = purchaseRequisitionData.DescItem;}  
else  
{desc = ""};}
```

```
var rrc;  
if(purchaseRequisitionData.ResponsavelRC)  
{rrc = purchaseRequisitionData.ResponsavelRC;}  
else  
{rrc="";}
```

```
var condp;  
if(purchaseRequisitionData.CondPgto)
```

```
{condp = purchaseRequisitionData.CondPgto;}
else
{condp = "";}


```

```
var con;
if(purchaseRequisitionData.Contrato)
{con = purchaseRequisitionData.Contrato;}
else
{con = "";}


```

```
var dtini;
if(purchaseRequisitionData.DataInicio)
{dtini = purchaseRequisitionData.DataInicio;}
else
{dtini = "";}


```

```
var dtfim;
if(purchaseRequisitionData.DataFim)
{dtfim = purchaseRequisitionData.DataFim;}
else
{dtfim="";}


```

```
//Contrato data
var cons;
if(purchaseRequisitionData.ContratoStatus)
{cons = purchaseRequisitionData.ContratoStatus;}
else
{cons="";}


```

```
var cono;
if(purchaseRequisitionData.ContratoOwner)
{cono = purchaseRequisitionData.ContratoOwner;}
else
{cono="";}


```

```
var state;
if(purchaseRequisitionData.STATE)
{state = purchaseRequisitionData.STATE;}
else
{state="";}


```

```
purchaseRequisitionDataArray.push({'TipoDocumento':"Compras",
'PO':po, 'DataNF':dtnf,
'DataPG':dtpg,'StatusNF':st,'MontanteNF':mon,'TipoCompra':tc,'SubtipoCompra':stc,'RC':rc,'DataRC':dtrc,'StatusRC':src,'Aprovador':ap,'DataPO':dtpo,'Buyer':by,'Departamento':dept,'Montante':monc,'Saldo':saldo,'NomeFornecedor':nfor,'CNPJFornecedor':cnpj,'Contrato':con,'DataInicio':dtini,'DataFim':dtfim,'ContratoStatus':cons,'ContratoOwner':cono,'Requisitante':req,'DescItem':desc,'ResponsavelRC':rrc, 'CondPgto':condp});
purchaseRequisitionDataArray.push({'STATE': state,
'TipoDocumento':"Compras",
'PO':po,'TipoCompra':tc,'SubtipoCompra':stc,'RC':rc,'DataRC':dtrc,'StatusRC':src,'Aprovador':ap,'DataPO':dtpo,'Buyer':by,'Departamento':dept,'Montante':monc,'Saldo':saldo,'NomeFornecedor':nfor,'C
```


NPJFornecedor':cnpj,'Contrato':con,'DataInicio':dtini,'DataFim':dtfim,'ContratoStatus':cons,'ContratoOwner':cono,'Requisitante':req,'DescItem':desc,'ResponsavelRC':rrc, 'CondPgto':condp});
//This piece of code has been updated cause we stored this information in the Purchase Requisition object thanks to the first query

```
queryNF(purchaseRequisitionData)
  .Then(function(successMsg){
    //Log.Verbose(successMsg);
    resolve("queryTABPO successful");
  }).Catch(function(err){
    reject(err);
  })

//Here
//.Then(function(resultOfAnotherQuery) {
//  //Reassign the result of another query in the purchaseRequisitionData object
//  //purchaseRequisitionData = resultOfAnotherQuery;
//  //resolve(purchaseRequisitionData)
//})
//.Catch(Log.Error)

} else {

    //PO data
    var po;
    if(purchaseRequisitionData.PO)
    {po = purchaseRequisitionData.PO;}
    else
    {po = "";}

    var dtpo;
    if(purchaseRequisitionData.DataPO)
    {dtpo = purchaseRequisitionData.DataPO; }
    else
    {dtpo="";}

    var by ;
    if(purchaseRequisitionData.Buyer)
    {by = purchaseRequisitionData.Buyer;}
    else
    {by="";}

    var saldo;
    if(purchaseRequisitionData.Saldo)
    {saldo = purchaseRequisitionData.Saldo; }
    else
    {saldo = 0;}
```

```
//NF data

var dtnf;
if(purchaseRequisitionData.DataNF)
{dtnf = purchaseRequisitionData.DataNF;}
else
{dtnf="";}

var dtpg;
if(purchaseRequisitionData.DataPG)
{dtpg = purchaseRequisitionData.DataPG;}
else
{dtpg="";}

var st;
if(purchaseRequisitionData.PassoNF)
{st = purchaseRequisitionData.PassoNF;}
else
{st="";}

var mon;
if(purchaseRequisitionData.MontanteNF)
{mon = purchaseRequisitionData.MontanteNF;}
else
{mon=0;}

//PR data
var tc;
if(purchaseRequisitionData.TipoCompra)
{tc = purchaseRequisitionData.TipoCompra; }
else
{tc="";}

var stc;
if(purchaseRequisitionData.SubtipoCompra)
{stc = purchaseRequisitionData.SubtipoCompra;}
else
{stc = "";}

var rc;
if(purchaseRequisitionData.RC)
{rc = purchaseRequisitionData.RC;}
else
{rc="";}

var dtrc;
if(purchaseRequisitionData.DataRC)
{dtrc = purchaseRequisitionData.DataRC;}
else
{dtrc="";}

var src;
```

```
if(purchaseRequisitionData.StatusRC)
{src =purchaseRequisitionData.StatusRC;}
else
{src="";}

var ap;
if(purchaseRequisitionData.Aprovador)
{ap = purchaseRequisitionData.Aprovador;}
else
{ap="";}

var dept;
if(purchaseRequisitionData.Departamento)
{dept = purchaseRequisitionData.Departamento;}
else
{dept="";}

var monc;
if(purchaseRequisitionData.Montante)
{monc = purchaseRequisitionData.Montante;}
else
{monc = 0;}

var nfor;
if(purchaseRequisitionData.NomeFornecedor)
{nfor = purchaseRequisitionData.NomeFornecedor;}
else
{nfor = "";}

var cnpj;
if(purchaseRequisitionData.CNPJFornecedor)
{cnpj = purchaseRequisitionData.CNPJFornecedor;}
else
{cnpj = "";}

var req;
if(purchaseRequisitionData.Requisitante)
{req = purchaseRequisitionData.Requisitante;}
else
{req = "";}

var desc;
if(purchaseRequisitionData.DescItem)
{desc = purchaseRequisitionData.DescItem;}
else
{desc = "";}

var rrc;
if(purchaseRequisitionData.ResponsavelRC)
{rrc = purchaseRequisitionData.ResponsavelRC;}
else
{rrc="";}
```

```
var condp;  
if(purchaseRequisitionData.CondPgto)  
{condp = purchaseRequisitionData.CondPgto;}  
else  
{condp = "";} 
```

```
var con;  
if(purchaseRequisitionData.Contrato)  
{con = purchaseRequisitionData.Contrato;}  
else  
{con = "";} 
```

```
var dtini;  
if(purchaseRequisitionData.DataInicio)  
{dtini = purchaseRequisitionData.DataInicio;}  
else  
{dtini = "";} 
```

```
var dtfim;  
if(purchaseRequisitionData.DataFim)  
{dtfim = purchaseRequisitionData.DataFim;}  
else  
{dtfim="";} 
```

```
//Contrato data  
var cons;  
if(purchaseRequisitionData.ContratoStatus)  
{cons = purchaseRequisitionData.ContratoStatus;}  
else  
{cons="";} 
```

```
var cono;  
if(purchaseRequisitionData.ContratoOwner)  
{cono = purchaseRequisitionData.ContratoOwner;}  
else  
{cono="";} 
```

```
var state;  
if(purchaseRequisitionData.STATE)  
{state = purchaseRequisitionData.STATE;}  
else  
{state="";} 
```

```
purchaseRequisitionDataArray.push({'STATE': state,  
"TipoDocumento":"Compras",  
'PO':po,'TipoCompra':tc,'SubtipoCompra':stc,'RC':rc,'DataRC':dtrc,'StatusRC':src,'Aprovador':ap,'DataPO':dtpo,'Buyer':by,'Departamento':dept,'Montante':monc,'Saldo':saldo,'NomeFornecedor':nfor,'CNPJFornecedor':cnpj,'Contrato':con,'DataInicio':dtini,'DataFim':dtfim,'ContratoStatus':cons,'ContratoOwner':cono,'Requisitante':req,'DescItem':desc,'ResponsavelRC':rrc, 'CondPgto':condp});
```

```

        resolve("error TABPO");
    }
} else {
    reject("error TABPO Move First");
}

});
}

function queryTabPR(purchaseRequisitionData) {
    return Sys.Helpers.Promise.Create(function (resolve, reject) {

        var tabletbpr = "AP - Purchase requisition - Headers__";
        var filtertbpr = "(OrderNumber__="+purchaseRequisitionData.RC+")";
        var QueryTPR = Process.CreateQueryAsProcessAdmin();
        QueryTPR.Reset();
        QueryTPR.SetSpecificTable(tabletbpr);
        QueryTPR.SetFilter(filtertbpr);
        QueryTPR.SetAttributesList("*");
        QueryTPR.SetSearchInArchive(true);
        if (QueryTPR.MoveFirst())
        {
            //Data.SetValue("Retorno__",QueryC.GetRecordCount());
            var transporttbpr = QueryTPR.MoveNextRecord();
            if (transporttbpr) {
                var varstbpr = transporttbpr.GetVars();
                var oa = 0.0;

                var ia = 0.0;

                if(varstbpr.GetValue_Double("OrderedAmount__",0))
                {

                    oa = varstbpr.GetValue_Double("OrderedAmount__",0);
                }

                if(varstbpr.GetValue_Double("InvoicedAmount__",0))
                {

                    ia = varstbpr.GetValue_Double("InvoicedAmount__",0);
                }

                //Log.Error("QA: " + oa);
                //Log.Error("IA: " + ia);

                purchaseRequisitionData.Saldo = (oa - ia);
                purchaseRequisitionData.Saldo =
purchaseRequisitionData.Saldo.toFixed(2);
                //Log.Verbose(purchaseRequisitionData.Saldo);
                //purchaseRequisitionData.Saldo = 999;
            }
        }
    });
}

```

```

purchaseRequisitionData.PO = "";
/*
//purchaseRequisitionDataArray.push(purchaseRequisitionData);
var po = "";//purchaseRequisitionData.PO;
var dtpo = "";//purchaseRequisitionData.DataPO; //Error
var by = "";//purchaseRequisitionData.Buyer; //Error

var saldo = purchaseRequisitionData.Saldo; //Error

var dtnf = "";
var dtpg = "";
var st = "";
var mon = "";

var tc = purchaseRequisitionData.TipoCompra;
var stc = purchaseRequisitionData.SubtipoCompra;
var rc = purchaseRequisitionData.RC
var dtrc = purchaseRequisitionData.DataRC;
var src = purchaseRequisitionData.StatusRC;
var ap = purchaseRequisitionData.Aprovador;
var dept = purchaseRequisitionData.Departamento;
var monc = purchaseRequisitionData.Montante;
var nfor = purchaseRequisitionData.NomeFornecedor;
var cnpj = purchaseRequisitionData.CNPJFornecedor;
var req = purchaseRequisitionData.Requisitante; //Error
var desc = purchaseRequisitionData.DescItem; //Error
var rrc = purchaseRequisitionData.ResponsavelRC; //Error
var condp = purchaseRequisitionData.CondPgto; //Error

var con = purchaseRequisitionData.Contrato;
var dtini = purchaseRequisitionData.DataInicio;
var dtfim = purchaseRequisitionData.DataFim;
var cons = purchaseRequisitionData.ContratoStatus;
var cono = purchaseRequisitionData.ContratoOwner;
*/

//PO data
var po;
if(purchaseRequisitionData.PO)
{po = purchaseRequisitionData.PO;}
else
{po = "";}

var dtpo;
if(purchaseRequisitionData.DataPO)
{dtpo = purchaseRequisitionData.DataPO; }
else
{dtpo="";}

```

```
var by ;
if(purchaseRequisitionData.Buyer)
{by = purchaseRequisitionData.Buyer;}
else
{by="";}
```

```
var saldo;
if(purchaseRequisitionData.Saldo)
{saldo = purchaseRequisitionData.Saldo; }
else
{saldo = 0;}
```

```
//NF data
```

```
var dtnf;
if(purchaseRequisitionData.DataNF)
{dtnf = purchaseRequisitionData.DataNF;}
else
{dtnf="";}
```

```
var dtpg;
if(purchaseRequisitionData.DataPG)
{dtpg = purchaseRequisitionData.DataPG;}
else
{dtpg="";}
```

```
var st;
if(purchaseRequisitionData.PassoNF)
{st = purchaseRequisitionData.PassoNF;}
else
{st="";}
```

```
var mon;
if(purchaseRequisitionData.MontanteNF)
{mon = purchaseRequisitionData.MontanteNF;}
else
{mon=0;}
```

```
//PR data
```

```
var tc;
if(purchaseRequisitionData.TipoCompra)
{tc = purchaseRequisitionData.TipoCompra; }
else
{tc="";}
```

```
var stc;
if(purchaseRequisitionData.SubtipoCompra)
{stc = purchaseRequisitionData.SubtipoCompra;}
else
{stc = "";}

```

```
var rc;
if(purchaseRequisitionData.RC)
{rc = purchaseRequisitionData.RC;}
else
{rc="";}

var dtrc;
if(purchaseRequisitionData.DataRC)
{dtrc = purchaseRequisitionData.DataRC;}
else
{dtrc="";}

var src;
if(purchaseRequisitionData.StatusRC)
{src =purchaseRequisitionData.StatusRC;}
else
{src="";}

var ap;
if(purchaseRequisitionData.Aprovador)
{ap = purchaseRequisitionData.Aprovador;}
else
{ap="";}

var dept;
if(purchaseRequisitionData.Departamento)
{dept = purchaseRequisitionData.Departamento;}
else
{dept="";}

var monc;
if(purchaseRequisitionData.Montante)
{monc = purchaseRequisitionData.Montante;}
else
{monc = 0;}

var nfor;
if(purchaseRequisitionData.NomeFornecedor)
{nfor = purchaseRequisitionData.NomeFornecedor;}
else
{nfor = "";}

var cnpj;
if(purchaseRequisitionData.CNPJFornecedor)
{cnpj = purchaseRequisitionData.CNPJFornecedor;}
else
{cnpj = "";}

var req;
if(purchaseRequisitionData.Requisitante)
{req = purchaseRequisitionData.Requisitante;}
else
```



```

{req = "";}

var desc;
if(purchaseRequisitionData.DescItem)
{desc = purchaseRequisitionData.DescItem;}
else
{desc = "";}

var rrc;
if(purchaseRequisitionData.ResponsavelRC)
{rrc = purchaseRequisitionData.ResponsavelRC;}
else
{rrc="";}

var condp;
if(purchaseRequisitionData.CondPgto)
{condp = purchaseRequisitionData.CondPgto;}
else
{condp = "";}

var con;
if(purchaseRequisitionData.Contrato)
{con = purchaseRequisitionData.Contrato;}
else
{con = "";}

var dtini;
if(purchaseRequisitionData.DataInicio)
{dtini = purchaseRequisitionData.DataInicio;}
else
{dtini = "";}

var dtfim;
if(purchaseRequisitionData.DataFim)
{dtfim = purchaseRequisitionData.DataFim;}
else
{dtfim="";}

//Contrato data
var cons;
if(purchaseRequisitionData.ContratoStatus)
{cons = purchaseRequisitionData.ContratoStatus;}
else
{cons="";}

var cono;
if(purchaseRequisitionData.ContratoOwner)
{cono = purchaseRequisitionData.ContratoOwner;}
else
{cono="";}

var state;

```

```

        if(purchaseRequisitionData.STATE)
        {state = purchaseRequisitionData.STATE;}
        else
        {state="";}

```

```

        purchaseRequisitionDataArray.push({'STATE' : state,
'TipoDocumento':"Compras",
'PO':po,'TipoCompra':tc,'SubtipoCompra':stc,'RC':rc,'DataRC':dtrc,'StatusRC':src,'Aprovador':ap,'D
ataPO':dtpo,'Buyer':by,'Departamento':dept,'Montante':monc,'Saldo':saldo,'NomeFornecedor':nfor,'C
NPJFornecedor':cnpj,'Contrato':con,'DataInicio':dtini,'DataFim':dtfim,'ContratoStatus':cons,'Contrat
oOwner':cono,'Requisitante':req,'DescItem':desc,'ResponsavelRC':rrc,'CondPgto':condp});
        //This piece of code has been updated cause we stored this information in the Purchase
Requisition object thanks to the first query

```

```

        queryNF(purchaseRequisitionData)
            .Then(function(successMsg){
                //Log.Verbose(successMsg);
                resolve("queryTABPR successful");
            }).Catch(function(err){
                reject(err);
            })

            //Here
            //.Then(function(resultOfAnotherQuery) {
            //Reassign the result of another query in the purchaseRequisitionData object
            //purchaseRequisitionData = resultOfAnotherQuery;
            //resolve(purchaseRequisitionData)
            //})
            //.Catch(Log.Error)

```

```

    } else {

        //PO data
        var po;
        if(purchaseRequisitionData.PO)
        {po = purchaseRequisitionData.PO;}
        else
        {po = "";}

        var dtpo;
        if(purchaseRequisitionData.DataPO)
        {dtpo = purchaseRequisitionData.DataPO; }
        else
        {dtpo="";}

        var by ;
        if(purchaseRequisitionData.Buyer)
        {by = purchaseRequisitionData.Buyer;}
        else
        {by="";}

```

```
var saldo;
if(purchaseRequisitionData.Saldo)
{saldo = purchaseRequisitionData.Saldo; }
else
{saldo = 0;}

//NF data

var dtnf;
if(purchaseRequisitionData.DataNF)
{dtnf = purchaseRequisitionData.DataNF;}
else
{dtnf="";}

var dtpg;
if(purchaseRequisitionData.DataPG)
{dtpg = purchaseRequisitionData.DataPG;}
else
{dtpg="";}

var st;
if(purchaseRequisitionData.PassoNF)
{st = purchaseRequisitionData.PassoNF;}
else
{st="";}

var mon;
if(purchaseRequisitionData.MontanteNF)
{mon = purchaseRequisitionData.MontanteNF;}
else
{mon=0;}

//PR data
var tc;
if(purchaseRequisitionData.TipoCompra)
{tc = purchaseRequisitionData.TipoCompra; }
else
{tc="";}

var stc;
if(purchaseRequisitionData.SubtipoCompra)
{stc = purchaseRequisitionData.SubtipoCompra;}
else
{stc = "";}

var rc;
if(purchaseRequisitionData.RC)
{rc = purchaseRequisitionData.RC;}
else
{rc="";}
```

```
var dtrc;
if(purchaseRequisitionData.DataRC)
{dtrc = purchaseRequisitionData.DataRC;}
else
{dtrc="";}

var src;
if(purchaseRequisitionData.StatusRC)
{src =purchaseRequisitionData.StatusRC;}
else
{src="";}

var ap;
if(purchaseRequisitionData.Aprovador)
{ap = purchaseRequisitionData.Aprovador;}
else
{ap="";}

var dept;
if(purchaseRequisitionData.Departamento)
{dept = purchaseRequisitionData.Departamento;}
else
{dept="";}

var monc;
if(purchaseRequisitionData.Montante)
{monc = purchaseRequisitionData.Montante;}
else
{monc = 0;}

var nfor;
if(purchaseRequisitionData.NomeFornecedor)
{nfor = purchaseRequisitionData.NomeFornecedor;}
else
{nfor = "";}

var cnpj;
if(purchaseRequisitionData.CNPJFornecedor)
{cnpj = purchaseRequisitionData.CNPJFornecedor;}
else
{cnpj = "";}

var req;
if(purchaseRequisitionData.Requisitante)
{req = purchaseRequisitionData.Requisitante;}
else
{req = "";}

var desc;
if(purchaseRequisitionData.DescItem)
{desc = purchaseRequisitionData.DescItem;}
```

```
else
{desc = "";}

var rrc;
if(purchaseRequisitionData.ResponsavelRC)
{rrc = purchaseRequisitionData.ResponsavelRC;}
else
{rrc="";}

var condp;
if(purchaseRequisitionData.CondPgto)
{condp = purchaseRequisitionData.CondPgto;}
else
{condp = "";}

var con;
if(purchaseRequisitionData.Contrato)
{con = purchaseRequisitionData.Contrato;}
else
{con = "";}

var dtini;
if(purchaseRequisitionData.DataInicio)
{dtini = purchaseRequisitionData.DataInicio;}
else
{dtini = "";}

var dtfim;
if(purchaseRequisitionData.DataFim)
{dtfim = purchaseRequisitionData.DataFim;}
else
{dtfim="";}

//Contrato data
var cons;
if(purchaseRequisitionData.ContratoStatus)
{cons = purchaseRequisitionData.ContratoStatus;}
else
{cons="";}

var cono;
if(purchaseRequisitionData.ContratoOwner)
{cono = purchaseRequisitionData.ContratoOwner;}
else
{cono="";}

var state;
if(purchaseRequisitionData.STATE)
{state = purchaseRequisitionData.STATE;}
else
{state=""};}
```

```

        purchaseRequisitiondataArray.push({'STATE':
state,'TipoDocumento':"Compras",
'PO':po,'TipoCompra':tc,'SubtipoCompra':stc,'RC':rc,'DataRC':dtrc,'StatusRC':src,'Aprovador':ap,'D
ataPO':dtpo,'Buyer':by,'Departamento':dept,'Montante':monc,'Saldo':saldo,'NomeFornecedor':nfor,'C
NPJFornecedor':cnpj,'Contrato':con,'DataInicio':dtini,'DataFim':dtfim,'ContratoStatus':cons,'Contrat
oOwner':cono,'Requisitante':req,'DescItem':desc,'ResponsavelRC':rrc, 'CondPgto':condp});
        resolve("error TABPR / filtertbpr:" + filtertbpr);
    }
    } else {
        reject("error TABPR Move First");
    }

});
}

```

```

function queryNF(purchaseRequisitionData) {
    return Sys.Helpers.Promise.Create(function (resolve, reject) {

        var tablenfpo = "CDNAME#Notas_Fiscais";
        var filternfpo = "";
        //purchaseRequisitionData.PO = "4500001300";
        if(purchaseRequisitionData.PO)
        {
            filternfpo = "(&(Pedido_de_Compras__="+purchaseRequisitionData.PO+")
(Validation__!=FIRST)(Validation__!=DESTINATION)(State!=300)(State!=400)(Deleted=0))";
        }
        else
        {
            filternfpo = "(&(Pedido_de_Compras__="+purchaseRequisitionData.RC+")
(Validation__!=FIRST)(Validation__!=DESTINATION)(State!=300)(State!=400)(Deleted=0))";
        }
        //Log.Verbose(filternfpo);
        var QueryNFPO = Process.CreateQueryAsProcessAdmin();
        QueryNFPO.Reset();
        QueryNFPO.SetSpecificTable(tablenfpo);
        QueryNFPO.SetFilter(filternfpo);
        QueryNFPO.SetAttributesList("*");
        QueryNFPO.SetSearchInArchive(true);
        QueryNFPO.SetOptionEx("Limit=2000");
        //QueryNFPO.AddAttribute("DataFile");
        QueryNFPO.MoveFirst();
        var transportnf;
        transportnf = QueryNFPO.MoveNext();
        if (transportnf)
        {
            var num = QueryNFPO.GetRecordCount();
            Log.Verbose("Numero de NFs " + num + " para " + filternfpo);

            while(transportnf)

```

```

{

    //Log.Verbose("NF2");

    var varsnf;
    varsnf = transportnf.GetUninheritedVars();

    //purchaseRequisitionData.TipoDocumento = "Nota Fiscal";
    if(varsnf.GetValue_String("Data_Fatura___",0))
    {
        purchaseRequisitionData.DataNF =
varsnf.GetValue_String("Data_Fatura___",0);
    }
    else
    {
        purchaseRequisitionData.DataNF = ""
    }

    if(varsnf.GetValue_String("Dt_Basica___",0))
    {
        purchaseRequisitionData.DataPG =
varsnf.GetValue_String("Dt_Basica___",0);
    }
    else
    {
        purchaseRequisitionData.DataPG = "";
    }

    if(varsnf.GetValue_String("Validation___",0))
    {
        purchaseRequisitionData.PassoNF =
varsnf.GetValue_String("Validation___",0);
    }
    else
    {
        purchaseRequisitionData.PassoNF="";
    }

    if(varsnf.GetValue_String("Valor_Total_Nota___",0))
    {
        purchaseRequisitionData.MontanteNF =
varsnf.GetValue_String("Valor_Total_Nota___",0);
    }
    else
    {
        purchaseRequisitionData.MontanteNF =0;
    }

    //Testes varsnf
    //purchaseRequisitionData.DataNF
    //purchaseRequisitionData.DataPG

```

```
//purchaseRequisitionData.PassoNF  
//purchaseRequisitionData.MontanteNF
```

```
var po;  
if(purchaseRequisitionData.PO)  
{po = purchaseRequisitionData.PO;}  
else  
{po = "";} 
```

```
var dtpo;  
if(purchaseRequisitionData.DataPO)  
{dtpo = purchaseRequisitionData.DataPO; }  
else  
{dtpo="";} 
```

```
var by ;  
if(purchaseRequisitionData.Buyer)  
{by = purchaseRequisitionData.Buyer;}  
else  
{by="";} 
```

```
var saldo;  
if(purchaseRequisitionData.Saldo)  
{saldo = purchaseRequisitionData.Saldo; }  
else  
{saldo = 0;} 
```

```
var dtmf;  
if(purchaseRequisitionData.DataNF)  
{dtmf = purchaseRequisitionData.DataNF;}  
else  
{dtmf="";} 
```

```
var dtpg;  
if(purchaseRequisitionData.DataPG)  
{dtpg = purchaseRequisitionData.DataPG;}  
else  
{dtpg="";} 
```

```
var st;  
if(purchaseRequisitionData.PassoNF)  
{st = purchaseRequisitionData.PassoNF;}  
else  
{st="";} 
```

```
var mon;  
if(purchaseRequisitionData.MontanteNF)  
{mon = purchaseRequisitionData.MontanteNF;}  
else
```



```
{mon=0;}

var tc;
if(purchaseRequisitionData.TipoCompra)
{tc = purchaseRequisitionData.TipoCompra; }
else
{tc="";}

var stc;
if(purchaseRequisitionData.SubtipoCompra)
{stc = purchaseRequisitionData.SubtipoCompra;}
else
{stc = "";}

var rc;
if(purchaseRequisitionData.RC)
{rc = purchaseRequisitionData.RC;}
else
{rc="";}

var dtrc;
if(purchaseRequisitionData.DataRC)
{dtrc = purchaseRequisitionData.DataRC;}
else
{dtrc="";}

var src;
if(purchaseRequisitionData.StatusRC)
{src =purchaseRequisitionData.StatusRC;}
else
{src="";}

var ap;
if(purchaseRequisitionData.Aprovador)
{ap = purchaseRequisitionData.Aprovador;}
else
{ap="";}

var dept;
if(purchaseRequisitionData.Departamento)
{dept = purchaseRequisitionData.Departamento;}
else
{dept="";}

var monc;
if(purchaseRequisitionData.Montante)
{monc = purchaseRequisitionData.Montante;}
else
{monc = 0;}

var nfor;
if(purchaseRequisitionData.NomeFornecedor)
```

```
{nfor = purchaseRequisitionData.NomeFornecedor;}  
else  
{nfor = "";}}
```

```
var cnpj;  
if(purchaseRequisitionData.CNPJFornecedor)  
{cnpj = purchaseRequisitionData.CNPJFornecedor;}  
else  
{cnpj = "";}}
```

```
var req;  
if(purchaseRequisitionData.Requisitante)  
{req = purchaseRequisitionData.Requisitante;}  
else  
{req = "";}}
```

```
var desc;  
if(purchaseRequisitionData.DescItem)  
{desc = purchaseRequisitionData.DescItem;}  
else  
{desc = "";}}
```

```
var rrc;  
if(purchaseRequisitionData.ResponsavelRC)  
{rrc = purchaseRequisitionData.ResponsavelRC;}  
else  
{rrc="";}}
```

```
var condp;  
if(purchaseRequisitionData.CondPgto)  
{condp = purchaseRequisitionData.CondPgto;}  
else  
{condp = "";}}
```

```
var con;  
if(purchaseRequisitionData.Contrato)  
{con = purchaseRequisitionData.Contrato;}  
else  
{con = "";}}
```

```
var dtini;  
if(purchaseRequisitionData.DataInicio)  
{dtini = purchaseRequisitionData.DataInicio;}  
else  
{dtini = "";}}
```

```
var dtfim;  
if(purchaseRequisitionData.DataFim)  
{dtfim = purchaseRequisitionData.DataFim;}  
else  
{dtfim="";}}
```

```

var cons;
if(purchaseRequisitionData.ContratoStatus)
{cons = purchaseRequisitionData.ContratoStatus;}
else
{cons="";}

var cono;
if(purchaseRequisitionData.ContratoOwner)
{cono = purchaseRequisitionData.ContratoOwner;}
else
{cono="";}

var state;
if(purchaseRequisitionData.STATE)
{state = purchaseRequisitionData.STATE;}
else
{state="";}

```

```

purchaseRequisitionDataArray.push({'STATE':state,'TipoDocumento':"Nota Fiscal", 'PO':po,
'DataNF':dtnf,
'DataPG':dtpg,'PassoNF':st,'MontanteNF':mon,'TipoCompra':tc,'SubtipoCompra':stc,'RC':rc,'DataRC':dtrc,'StatusRC':src,'Aprovador':ap,'DataPO':dtpo,'Buyer':by,'Departamento':dept,'Montante':monc,'Saldo':saldo,'NomeFornecedor':nfor,'CNPJFornecedor':cnpj,'Contrato':con,'DataInicio':dtini,'DataFim':dtfim,'ContratoStatus':cons,'ContratoOwner':cono,'Requisitante':req,'DescItem':desc,'ResponsaveIRC':rrc, 'CondPgto':condp});

```

```

//purchaseRequisitionDataArray.push({'TipoDocumento':"TESTENF"});

```

```

//purchaseRequisitionDataArray.push(purchaseRequisitionData);
transportnf = QueryNFPO.MoveNext();
}

```

```

resolve("Query NF successful");

```

```

} else {
//reject("Nao achou NF: " + filternfpo);
resolve("Query NF nao encontrou / filternfpo:" + filternfpo);
}

```

```

});
}

```

```

function BuildMyEmail(data)
{

```

```

if(data.length > 1)
{
    var fcontent="";
    for(var y=0;y<data.length;y++)
    {
        fcontent = fcontent + data[y].TipoDocumento + ";";

        if(data[y].RC)
        {
            fcontent = fcontent + data[y].RC + ";";
        }
        else
        {
            fcontent = fcontent + ";";
        }

        //if(data[y].STATE)
        //{
        //fcontent = fcontent + data[y].STATE + ";";
        //}
        //else
        //{
        //fcontent = fcontent + ";";
        //}

        if(data[y].TipoCompra)
        {
            fcontent = fcontent + data[y].TipoCompra + ";";
        }
        else
        {
            fcontent = fcontent + ";";
        }

        if(data[y].SubtipoCompra)
        {
            fcontent = fcontent + data[y].SubtipoCompra + ";";
        }
        else
        {
            fcontent = fcontent + ";";
        }

        if(data[y].DataRC)
        {
            fcontent = fcontent + data[y].DataRC + ";";
        }
        else
        {

```

```

fcontent = fcontent + ";";
}

if(data[y].Requisitante)
{
fcontent = fcontent + data[y].Requisitante + ";";
}
else
{
fcontent = fcontent + ";";
}

if(data[y].StatusRC)
{
fcontent = fcontent + data[y].StatusRC + ";";
}
else
{
fcontent = fcontent + ";";
}
if(data[y].STATE)
{
fcontent = fcontent + data[y].STATE + ";";
}
else
{
fcontent = fcontent + ";";
}

if(data[y].ResponsavelRC && data[y].StatusRC !=
"Completed")

{
fcontent = fcontent + data[y].ResponsavelRC + ";";
}
else
{
fcontent = fcontent + ";";
}

if(data[y].DescItem)
{
fcontent = fcontent + data[y].DescItem.replace(/[/;]/g,"")
+ ";";
}
else
{
fcontent = fcontent + ";";
}

if(data[y].CondPgto)
{

```

```
r;]/g, ") + ";;";
```

```
fcontent = fcontent + data[y].CondPgto.replace(/[\n\
```

```
}
```

```
else
```

```
{
```

```
fcontent = fcontent + ";;";
```

```
}
```

```
if(data[y].Aprovador)
```

```
{
```

```
fcontent = fcontent + data[y].Aprovador + ";;";
```

```
}
```

```
else
```

```
{
```

```
fcontent = fcontent + ";;";
```

```
}
```

```
if(data[y].PO)
```

```
{
```

```
fcontent = fcontent + data[y].PO + ";;";
```

```
}
```

```
else
```

```
{
```

```
fcontent = fcontent + ";;";
```

```
}
```

```
if(data[y].DataPO)
```

```
{
```

```
fcontent = fcontent + data[y].DataPO + ";;";
```

```
}
```

```
else
```

```
{
```

```
fcontent = fcontent + ";;";
```

```
}
```

```
if(data[y].Buyer)
```

```
{
```

```
fcontent = fcontent + data[y].Buyer + ";;";
```

```
}
```

```
else
```

```
{
```

```
fcontent = fcontent + ";;";
```

```
}
```

```
if(data[y].Departamento)
```

```
{
```

```
fcontent = fcontent + data[y].Departamento + ";;";
```

```
}
```

```
else
```

```
{
```

```
fcontent = fcontent + ";;";
```

```
}
```

```

        if(data[y].Montante)
        {
            //fcontent = fcontent + data[y].Montante.replace(".",",")
+ ";;

        if(y>0)
        {

            var mt =
parseFloat(data[y].Montante).toLocaleString('de-DE', {maximumFractionDigits : 2,
minimumFractionDigits: 2});

            fcontent = fcontent + mt + ";;

        }
        else
        {
            fcontent = fcontent + data[y].Montante +
";";
        }
    }
    else
    {
        fcontent = fcontent + ";;
    }

    if(data[y].Saldo)
    {
        //fcontent = fcontent + data[y].Saldo.replace(".",",") +
";";

        if(y>0)
        {

            var mt =
parseFloat(data[y].Saldo).toLocaleString('de-DE', {maximumFractionDigits : 2,
minimumFractionDigits: 2});

            fcontent = fcontent + mt + ";;

        }
        else
        {
            fcontent = fcontent + data[y].Saldo + ";;

        }
    }
    else
    {
        fcontent = fcontent + ";;
    }

```

```

        if(data[y].NomeFornecedor)
        {
            fcontent = fcontent + data[y].NomeFornecedor + ";";
        }
        else
        {
            fcontent = fcontent + ";";
        }

        if(data[y].CNPJFornecedor)
        {
            //fcontent = fcontent + data[y].CNPJFornecedor + ";";
            if(y>0)
            {
                if(data[y].CNPJFornecedor.length > 11 )
                {
                    fcontent = fcontent + " " +
mascaraCnpj(data[y].CNPJFornecedor) + " " +";";
                }
                else
                {
                    fcontent = fcontent + " " +
mascaraCpf(data[y].CNPJFornecedor) + " " +";";
                }
            }
            else
            {
                fcontent = fcontent + " " +
data[y].CNPJFornecedor + " " +";";
            }
        }
        else
        {
            fcontent = fcontent + ";";
        }

        if(data[y].Contrato)
        {
            fcontent = fcontent + data[y].Contrato + ";";
        }
        else
        {
            fcontent = fcontent + ";";
        }

        if(data[y].ContratoStatus)
        {
            fcontent = fcontent + data[y].ContratoStatus + ";";
        }
        else
        {

```



```

fcontent = fcontent + ";";
}

if(data[y].ContratoOwner)
{

fcontent = fcontent + data[y].ContratoOwner + ";";
}
else
{
fcontent = fcontent + ";";
}

if(data[y].DataInicio)
{
fcontent = fcontent + data[y].DataInicio + ";";
}
else
{
fcontent = fcontent + ";";
}

if(data[y].DataFim)
{
fcontent = fcontent + data[y].DataFim + ";";
}
else
{
fcontent = fcontent + ";";
}

if(data[y].DataNF)
{
fcontent = fcontent + data[y].DataNF + ";";
}
else
{
fcontent = fcontent + ";";
}

if(data[y].DataPG)
{
fcontent = fcontent + data[y].DataPG + ";";
}
else
{
fcontent = fcontent + ";";
}

if(data[y].PassoNF)
{
fcontent = fcontent + data[y].PassoNF + ";";
}

```

```

    }
    else
    {
        fcontent = fcontent + ";";
    }

    if(data[y].MontanteNF)
    {

        //fcontent = fcontent +
data[y].MontanteNF.replace(".",",") + "\n";
        if(y>0)
        {

            var mt =
parseFloat(data[y].MontanteNF).toLocaleString('de-DE', {maximumFractionDigits : 2,
minimumFractionDigits: 2});

            fcontent = fcontent + mt + "\n";

        }
        else
        {
            fcontent = fcontent +
data[y].MontanteNF + "\n";
        }
    }
    else
    {
        fcontent = fcontent + "\n";
    }
}
Log.Info("Montou arquivo: " + data.length);
var tempFile = TemporaryFile.CreateFile("csv", "utf8");
TemporaryFile.Append(tempFile, fcontent);

var at = Attach.AttachTemporaryFile(tempFile,
"Relatorio.csv");

//Data.SetValue("Numero__",numero);
Log.Error("Loops" + loops);
Log.Error("Size: " + data.length);

if(Data.GetValue("PorEmail__")==="1")
{

    var emailMessage = "Ola!\n\n";
    emailMessage += "Seu relatorio solicitado de compras
esta em anexo neste email.\n";
    emailMessage += "Dados:\n\n";

```

```

        emailMessage += "Departamento: " +
Data.GetValue("Departamento__") + "\n\n";

        var dti = new Date(Data.GetValue("Inicio__"));
        var dtf = new Date(Data.GetValue("Fim__"));

        var inicio = (adicionaZero(dti.getDate().toString()) +
"/" + (adicionaZero(dti.getMonth()+1).toString()) + "/" + dti.getFullYear());
        var final = (adicionaZero(dtf.getDate().toString()) + "/"
+ (adicionaZero(dtf.getMonth()+1).toString()) + "/" + dtf.getFullYear());
        emailMessage += "Data de inicio: " + inicio + "\n\n";
        emailMessage += "Data de fim: " + final + "\n\n";

        emailMessage += "Atenciosamente,\n";
        emailMessage += "Administração";

        enviarEmail("Relatorio de
Compras",emailMessage,Data.GetValue("Email__"),at);
        Process.LeaveForm();
    }

    }
    //Log.Verbose(outputPurchaseRequisitionDataArray.length);

}
function enviarEmail(subject, emailMessage, email_RecipientAddress,file){

    // Create email transport and define its base properties
    var email = Process.CreateTransport("mail");
    var emailVars = email.GetUninheritedVars();
    var filex = Attach.GetInputFile(Attach.GetNbAttach()-1);
    if(filex)
    {
        var x = email.AddAttachEx(filex);
        var xaVars = x.GetVars();
        xaVars.AddValue_String("AttachOutputName", "Relatório", true);
    }
    else
    {
        var emailMessage = "Ola!\n\n";
        emailMessage += "Não foram encontrados documentos de compras com as
referencias solicitadas.\n";
        emailMessage += "Dados:\n\n";
        emailMessage += "Departamento: " + Data.GetValue("Departamento__") + "\n\n";

        var dti = new Date(Data.GetValue("Inicio__"));
        var dtf = new Date(Data.GetValue("Fim__"));
    }
}

```

```

        var inicio = (adicionaZero(dti.getDate().toString()) + "/" +
(adicionaZero(dti.getMonth()+1).toString()) + "/" + dti.getFullYear());
        var final = (adicionaZero(dtf.getDate().toString()) + "/" +
(adicionaZero(dtf.getMonth()+1).toString()) + "/" + dtf.getFullYear());
        emailMessage += "Data de inicio: " + inicio + "\n\n";
        emailMessage += "Data de fim: " + final + "\n\n";
        emailMessage += "Atenciosamente,\n";
        emailMessage += "Administração";

    }

    //at.SetAttachFile(file.GetFileName());
    // Common transport variables
    emailVars.AddValue_String("Subject", subject , true);
    emailVars.AddValue_String("Message", emailMessage, true);

    // Transport type specific variable: EmailAdress is required for email transports
    emailVars.AddValue_String( "EmailAddress", email_RecipientAddress, true);

    // Recipient level inherited variable: This is the email address that appears in the sent mail
    emailVars.AddValue_String( "FromAddress", "noreply.bmb@h3talent.com.br", true);

    // Add a Sender to the email and set his name
    var sender = email.SetSender();
    sender.GetVars().AddValue_String("FromName", "Relatório de Compras", true);
    Log.Info("Email enviado para: " + email_RecipientAddress);
    // Submit the email to the server
    email.Process();

}

function replaceAll(str, find, replace) {
    var r="";
    while(r!=str){
        r = str;
        str = str.replace(find, replace);
    }
    return str;
}

function zeroFill( number, width )
{
    width -= number.toString().length;
    if ( width > 0 )
    {
        return new Array( width + (/\./.test( number ) ? 2 : 1 ) ).join( '0' ) + number;
    }
    number = number + "";

```

```
    return number.toString().substring(0,10); // always return a string
}

function adicionaZero(numero){
    if (numero <= 9)
        return "0" + numero;
    else
        return numero;
}

function mascaraCpf(valor) {
    return valor.replace(/(\d{3})(\d{3})(\d{3})(\d{2})/g,"$1.$2.$3-$4");
}

function mascaraCnpj(valor) {
    return valor.replace(/(\d{2})(\d{3})(\d{3})(\d{4})(\d{2})/g,"$1.$2.$3\-$4-$5");
}
```