**1.1 US#1234**

**Configuration Attributes**

|  |  |  |
| --- | --- | --- |
| Attribute Name | Variable Name | Description |
| Region | Region\_pcr | Used to show the region. |
| Blades | blades\_quantity | Created to show the blades quantity. |

**Recommendation Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Packaged Solutions [Table] | packagedSolutions | This rec rule is used to package the solutions. |
| Populate Blade Image [Easy Advanced] | populateBladeEnclosureImage | This rule is used to populate the image of blade in the page. |

**Constraint Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Constrain Chassis by Performance [Table] | constrainChassisByPerformance | This rule constraints the chassis by performance levels. |

**Hiding Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Hide Blade Quantity/Enclosure [Simple] | hideBladeQty | This rule hides the blade quantity/enclosure |

**BML Util Libraries**

|  |  |  |
| --- | --- | --- |
| Util Name | Variable Name | Description |
| Parse CSV | parseCSV | This util is used to parse the CSV file. |

**Script Text**

arr = string[][];  
raw = urldatabyget(fileLocation, "", "ERROR");  
if(raw == "ERROR"){
return arr;  
}
goodLines = 0;  
knownColumns = 0;  
rawLines = split(raw,"\n");  
for rl in rawLines{
rawCols = split(rl,",");  
if(goodLines == 0){
knownColumns = sizeofarray(rawCols);  
}
if(sizeofarray(rawCols) <> knownColumns){
break;  
}
else{
goodLines = goodLines + 1;  
}
}
rows = range(goodLines);  
cols = range(knownColumns);  
for row in rows{
rawColumns = split(rawLines[row],",");  
for col in cols{
arr[row][col] = rawColumns[col];  
}
}
return arr;

**BML Util Libraries**

|  |  |  |
| --- | --- | --- |
| Util Name | Variable Name | Description |
| Quote Analysis PL | quoteAnalysisPL | This util analyze the quote PL. |

**Script Text**

result="";  
monthlyRate = pow( 1 + (discountRate / 100.0), 1 / 12.0);  
term = range(6);  
decimalPlace = 2;  
 // round values to this many decimal places
for t in term{
years = t;  
months = 12 \* years;  
nominalRevenue = oneTimeRevenue + (months \* monthlyRevenue);  
nominalCost = oneTimeCost + (months \* monthlyCost);  
nominalPL = nominalRevenue - nominalCost;  
nominalReturn = 0.0;  
if(nominalCost <> 0.0){
nominalReturn = ((nominalRevenue / nominalCost) - 1.00) \* 100.0;  
nominalReturn = round(nominalReturn, decimalPlace);  
 // added on 2014.12.29 to resolve rounding error that emerged after the 2014R2 upgrade
}
counter = range(months);  
NPVRevenue = oneTimeRevenue;  
NPVCost = oneTimeCost;  
for c in counter{
NPVRevenue = NPVRevenue + monthlyRevenue / pow(monthlyRate, c + 1.0);  
NPVCost = NPVCost + monthlyCost / pow(monthlyRate, c + 1.0);  
}
NPVPL = NPVRevenue - NPVCost;  
NPVreturn = 0.0;  
if(NPVCost <> 0.0){
NPVreturn = ((NPVRevenue / NPVCost) - 1.00) \* 100.0;  
NPVreturn = round(NPVreturn, decimalPlace);  
 // added on 2014.12.29 to resolve rounding error that emerged after the 2014R2 upgrade
}
if(t == 0){
result = result + "1~nominalRevenue\_t~" + string (nominalRevenue) + "|" ;  
result = result + "1~nominalCost\_t~" + string (nominalCost) + "|";  
result = result + "1~nominalPL\_t~" + string (nominalPL) + "|";  
result = result + "1~nominalReturn\_t~" + string (nominalReturn) + "|";  
result = result + "1~nPVRevenue\_t~" + string (NPVRevenue) + "|";  
result = result + "1~nPVCost\_t~" + string (NPVCost) + "|";  
result = result + "1~nPVPL\_t~" + string (NPVPL) + "|";  
result = result + "1~nPVReturn\_t~" + string (NPVreturn) + "|";  
}
else{
result = result + "1~year" + string(t) + "NominalRevenue\_t~" + string (nominalRevenue) + "|";  
result = result + "1~year" + string(t) + "NominalCost\_t~" + string (nominalCost) + "|";  
result = result + "1~year" + string(t) + "NominalPL\_t~" + string (nominalPL) + "|";  
result = result + "1~year" + string(t) + "NominalReturn\_t~" + string (nominalReturn) + "|";  
result = result + "1~year" + string(t) + "NPVRevenue\_t~" + string (NPVRevenue) + "|";  
result = result + "1~year" + string(t) + "NPVCost\_t~" + string (NPVCost) + "|";  
result = result + "1~year" + string(t) + "NPVPL\_t~" + string (NPVPL) + "|";  
result = result + "1~year" + string(t) + "NPVReturn\_t~" + string (NPVreturn) + "|";  
}
}
return result;

**1.2 US#5678**

**Configuration Attributes**

|  |  |  |
| --- | --- | --- |
| Attribute Name | Variable Name | Description |
| Enclosures Required | enclosuresRequired | Each Vision Blade 6000 has a maximum capacity of 10 Blades |
| Enclosure | enclosure | This attribute holds the enclosure details |

**Recommendation Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Set Enclosures Required [Easy Advanced] | setEnclosuresRequired | This rec rule is used to set the required enclosures. |
| Force Set Chassis [Simple] | forceSetChassis | This rule force set the chassis. |

**Constraint Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Constrain Rack by Size [Simple/Table] | constrainRackBySize | This rule constraints the rack size if it crosses the limits. |

**Hiding Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Hide ServerQty/Rack [Simple] | hideServerQuantity | This rule hides the server quantity/racks |

**BML Util Libraries**

|  |  |  |
| --- | --- | --- |
| Util Name | Variable Name | Description |
| getBasicAuthCredentials | CN\_GetBasicAuthCredentials | This is used to retrieve the authorization credentials. |

**Script Text**

//get credentials for JCS-SXs 2 legged OAuth client that was stored by you in secure columns of custom data tables
clientname = accountName;  
clients = bmql("select username, password from CN\_Credentials where integrationName=$clientname");  
//base 64 encode the credentials, per Basic Authorization standard
for client in clients {
encodedCreds= encodebase64( get(client, "username") + ":" + get(client, "password"));  
return encodedCreds;  
}
//allow appropriate message to user, if integration client information was not located
return defaultReturnValue;

**BML Util Libraries**

|  |  |  |
| --- | --- | --- |
| Util Name | Variable Name | Description |
| Get File Attachment Data | CN\_GetFileAttachmentData | This is used to retrieve the file attachment. |

**Script Text**

response = dict("string");  
url = siteUrl+"/rest/v2"+
"/commerceProcesses/"+ processVarName+
"/transactions/"+ string(transactionId)+
"/attachments/"+ attachmentVarName+"/";  
headers = dict("string");  
put(headers, "Authorization", userAuth );  
put(headers, "Accept", "application/json");  
// call rest method
xlsxResponse = urldata( url, "GET", headers );  
if( get(xlsxResponse, "Status-Code") <> "200" ){
put(response, "error", get(xlsxResponse, "Error-Message"));  
return response;  
/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/
}else{
mb = get(xlsxResponse, "Message-Body");  
jmb = json(mb);  
fileContent = jsonget(jmb, "fileContent");  
if( len(fileContent) < 5 ){
put(response, "error", "GetFileAttachment: Could not locate document in Response.");  
return response;  
}
put(response, "base64doc", fileContent);  
return response;  
};

**1.3 US#7661**

**Configuration Attributes**

|  |  |  |
| --- | --- | --- |
| Attribute Name | Variable Name | Description |
| Power Supply | powerSupply | This menu attribute lists the power supply PSU's to choose. |

**Recommendation Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Set Remaining Blades [Easy Advanced] | setRemainingBlades | This rec rule helps to set the remaining blades slots. |

**Constraint Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Inventory Check [Simple] | inventoryCheck | Filler Rule - To be reimplemented with real inventory management with EBS in the future. |

**Hiding Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Hide Adv Rack Configuration [Simple] | hideAdvancedRackConfiguration | This rule hides the advance rack configuration if server module equals UPS 1500 |

**BML Util Libraries**

|  |  |  |
| --- | --- | --- |
| Util Name | Variable Name | Description |
| Get File Id From Response | CN\_GetFileIdFromResponse | This util is used to get the file ID from EBS response. |

**Script Text**

mb = get(response, "Message-Body");  
jmb = json(mb);  
ids = jsonget(jmb, "fileId");  
if( isnumber(ids) ){
return atoi(ids);  
}
return -1;

**BML Util Libraries**

|  |  |  |
| --- | --- | --- |
| Util Name | Variable Name | Description |
| Update File Attachment | CN\_UpdateFileAttachment | This util is used to update the file attachment. |

**Script Text**

response = dict("string");  
url = siteUrl+"/rest/v2"+
"/commerceProcesses/" + processVarName+
"/transactions/" + string(transactionId) +
"/attachments/" + attachmentVarname;  
headers = dict("string");  
put(headers, "Authorization", userAuth );  
put(headers, "Accept", "application/json");  
put(headers, "Content-Type", "application/json");  
// add json data
usePrintContext = "false";  
if(NOT hasBrowse){
usePrintContext = "true";  
}
jsonObj = json();  
jsonput(jsonObj, "mediaType", fileType);  
jsonput(jsonObj, "fileContent", fileContent);  
jsonput(jsonObj, "useTempTable", usePrintContext);  
jsonput(jsonObj, "usePrintContext", usePrintContext);  
jsonput(jsonObj, "fileName", fileName);  
// call rest method
xlsxResponse = urldata( url, "POST", headers, jsontostr(jsonObj) );  
if( get(xlsxResponse, "Status-Code") <> "200" ){
put(response, "error", get(xlsxResponse, "Error-Message"));  
return response;  
}else{
return xlsxResponse;  
};

**1.4 US#8341**

**Configuration Attributes**

|  |  |  |
| --- | --- | --- |
| Attribute Name | Variable Name | Description |
| Processor | processors | This menu attribute shows the list of processors available |

**Recommendation Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Set Total Equipment [Easy Advanced] | setTotalEquipment | This rule sets the total equipment used. |

**Constraint Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Contrain Processor\RAM\HDD\PSU\OS [Table] | contrainRAM | Based on performance this rule constraints processor/RAM/HDD/PSU/OS |

**Hiding Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Hide HDD Array [Simple] | hideHDDArray | This rule hides the HDD array if additional HDD's is false |

**BML Util Libraries**

|  |  |  |
| --- | --- | --- |
| Util Name | Variable Name | Description |
| Print Email Designer BML | CN\_PrintEmailDesigner | This util design the email format as approved by client. |

**Script Text**

// Prints a Email Designer Template. Returns "error" or "subject" & "body".
response = dict("string");  
/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/
// PRINT TEMPLATE
url = siteUrl+"/rest/v2/emailGenerator/";  
headers = dict("string");  
put(headers, "Authorization", userAuth );  
put(headers, "Accept", "application/json");  
put(headers, "Content-Type", "application/json");  
// add json data
jsonObj = json();  
jsonput(jsonObj, "processVarname", processVarName);  
jsonput(jsonObj, "templateName", templateName);  
jsonput(jsonObj, "transactionId", transactionId);  
jsonput(jsonObj, "languageCode", templateLanguage);  
// call rest method
xlsxResponse = urldata( url, "POST", headers, jsontostr(jsonObj) );  
// CHECK OUTPUT, UPDATE STATUS
statusCode = get(xlsxResponse, "Status-Code");  
printStatus = "";  
if( statusCode <> "204" ){
printStatus = "Print FAILED: " + templateName + "\n" +
"Error: " + get(xlsxResponse, "Error-Message");  
put(response, "error", printStatus);  
return response;  
/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/
}else{
printStatus = "Print Completed: " + templateName + "\n";  
}
/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/
// GET PRINT OUTPUT
url = get(xlsxResponse, "Location");  
headers = dict("string");  
put(headers, "Authorization", userAuth);  
put(headers, "Accept", "application/json");  
// get print output call
xlsxResponse = urldata( url, "GET", headers );  
mb = get(xlsxResponse, "Message-Body");  
if(len(mb) > 0){
jmb = json(mb);  
printOutput = jsonget(jmb , "subject" );  
put(response, "subject", printOutput);  
printOutput = jsonget(jmb , "body" );  
put(response, "body", printOutput);  
}else{
put(response, "error", "Email Output was empty.");  
}
print("PRINT COMPLETE");  
print("");  
return response;

**1.5 US#1003**

**Configuration Attributes**

|  |  |  |
| --- | --- | --- |
| Attribute Name | Variable Name | Description |
| RAM | ram | This menu attribute shows the list of RAM available |

**Recommendation Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Rec Cloud Backup [Simple] | recCloudBackup | This rec rule helps to protect the data using cloud backup |

**Constraint Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Open Source Software [Simple] | openSourceSoftware | This rule constraints OSS system if it requires Titanium support. |

**Hiding Rule**

|  |  |  |
| --- | --- | --- |
| Rule Name | Variable Name | Description |
| Hide Cloud Storage [Simple] | hideCloudBackup | This rule hides the cloud storage if cloud backup is false |

**BML Util Libraries**

|  |  |  |
| --- | --- | --- |
| Util Name | Variable Name | Description |
| Compare Docx | CN\_CompareDocx | This util helps to compare the docx files. |

**Script Text**

response = dict("string");  
// GENERATE DIFF REST CALL
url = siteUrl+"/rest/v2/docxCompare";  
jsonObj = json();  
jsonput(jsonObj, "processVarname", processVarName);  
jsonput(jsonObj, "transactionId", transactionId);  
jsonput(jsonObj, "oldDocAttachId", prevFileAttachId);  
jsonput(jsonObj, "newDocAttachId", curFileAttachId);  
headers = dict("string");  
put(headers, "Authorization", userAuth );  
put(headers, "Accept", "application/json");  
put(headers, "Content-Type", "application/json");  
// call rest method
xlsxResponse = urldata( url, "POST", headers, jsontostr(jsonObj) );  
if( get(xlsxResponse, "Status-Code") <> "200" ){
err = get(xlsxResponse, "Error-Message");  
if(len(err) <= 5){
err = get(xlsxResponse, "Message-Body");  
}
put(response, "error", err);  
return response;  
}else{
mb = get(xlsxResponse, "Message-Body");  
jmb = json(mb);  
diffs = jsonget(jmb , "diffs" );  
if(len(diffs) > 3){
put(response, "jsonString", mb);  
}
}
print("COMPARE COMPLETE");  
return response;

**BML Util Libraries**

|  |  |  |
| --- | --- | --- |
| Util Name | Variable Name | Description |
| Print Doc Designer | CN\_PrintDocDesigner | This util prints the doc designer template. |

**Script Text**

// Prints a Document Designer Document. Returns "error" or "base64doc".
response = dict("string");  
/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/
// PRINT TEMPLATE
url = siteUrl+"/rest/v2/documentGenerator/";  
headers = dict("string");  
put(headers, "Authorization", userAuth );  
put(headers, "Accept", "application/json");  
put(headers, "Content-Type", "application/json");  
// add json data
jsonObj = json();  
jsonput(jsonObj, "processVarname", processVarName);  
jsonput(jsonObj, "templateName", templateName);  
jsonput(jsonObj, "transactionId", transactionId);  
jsonput(jsonObj, "languageCode", templateLanguage);  
jsonput(jsonObj, "outputFormat", outputFormat);  
// call rest method
xlsxResponse = urldata( url, "POST", headers, jsontostr(jsonObj) );  
/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/
// CHECK OUTPUT, UPDATE STATUS
statusCode = get(xlsxResponse, "Status-Code");  
printStatus = "";  
if( statusCode <> "204" ){
printStatus = "Print FAILED: " + templateName + "\n" +
"Error: " + get(xlsxResponse, "Error-Message");  
put(response, "error", printStatus);  
return response;  
/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/
}else{
printStatus = "Print Completed: " + templateName + "\n";  
}
/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/
// GET PRINT OUTPUT
url = get(xlsxResponse, "Location");  
headers = dict("string");  
put(headers, "Authorization", userAuth);  
put(headers, "Accept", "application/json");  
// get print output call
xlsxResponse = urldata( url, "GET", headers );  
mb = get(xlsxResponse, "Message-Body");  
if(len(mb) > 0){
jmb = json(mb);  
printOutput = jsonget(jmb , "document" );  
put(response, "base64doc", printOutput);  
}else{
put(response, "error", "Document Output was empty.");  
}
print("PRINT COMPLETE");  
print("");  
return response;

**Data Table List**

|  |  |
| --- | --- |
| Data Table Name | Data Table Columns |
| Status | category , action , id , StatusOSC\_t , StatusLabel , StatusOSC\_l , HistoricalStatus , StatusID |
| osImages | category , action , id , operatingSystem , imageURL |
| monitoringPricing | category , action , id , Monitoring , pricetype , priceValue |
| ServerModules | category , action , id , Part , Module |
| ServerPerformance | category , action , id , Performance , Chassis , Processor , RAM , HDD , PSU , OperatingSystem |
| osVolumePricing | category , action , id , OS , Part , Comment , minQty , maxQty , Price |
| Processors\_ | category , action , id , Part , Processor |
| Packages | category , action , id , Package , Performance , Chassis , PSU , Processor , RAM , RamConfig , HD , OperatingSystem , NumProcessors |
| Support\_v | category , action , id , Part , SupportLevel , basePrice |
| SoftwarePackage | category , action , id , Part , Software , LicensePrice , MonthlyMaint |
| ServiceDetails | category , action , id , Part , Resource , Service , Rate , Price |
| PowerSupply | category , action , id , Part , PowerSupply |
| HardDrives | category , action , id , Part , HardDrive |
| Services | category , action , id , Service , ServiceDetail |
| CloudBackup | category , action , id , Part\_FeeType , FeeType , Part\_StorageAmt , CloudSpace , Comment |
| RAM | category , action , id , Part , RAM |
| RackSize | category , action , id , Part , Rack , MaxSize |
| INT\_SYSTEM\_DETAILS | category , action , id , System , Username , Endpoint , MaxLinesInPayload , Password |
| INT\_SYSTEM\_TEMPLATES | category , action , id , System , Operation , Template |
| LandingPages | category , action , id , Model , HTML , Mobile |
| PeriodConversations | category , action , id , custom\_part\_field4 , FusionValue , FusionLabel |
| PERIODIC\_CONVERSIONS | category , action , id , PartField , FusionValue , FusionLabel |
| MODEL\_CONVERSIONS | category , action , id , ModelLabel , ModelVarName , ItemId , OrgId , FeeType , Family , Cost |
| LandingPagesMobile | category , action , id , Model , HTML |
| RateFactor | category , action , id , Term , Rate , Yield |
| Financing | category , action , id , FinOptions , Term , Interest |
| pricing | category , action , id , entityId , state , entitytype , part , price |
| TestHAr | category , action , id , Test |
| CountryDetails | category , action , id , CountryName , State , Address , Phone , ZIpcode , Image , Answer , Question |
| DataToolAuth | category , action , id , UserName , Access |
| customerDetails | category , action , id , name , age , address , phoneNo |
| companyDetails | category , action , id , comapanyName , companyAddress , TelNumber , FaxNumber |
| employeeDetails | category , action , id , employeeName , Location , EmpId , ContactNumber |
| ModelFeatures | category , action , id , Frequency\_Band , Display\_Type , Mode |
| LanguageTranslations | category , action , id , English , French |
| ChicagoDataCenter | category , action , id , Name , Capacity , Serial\_Number |
| AryTable | category , action , id , Type , Country , State , City |
| Men\_Clothing | category , action , id , ModelNumber , Image |
| Customer | category , action , id , Name , LifeInsurance , Income , LoanRequested |
| Part | category , action , id , processor , operatingsystem , memory , software , Discount |
| PartPrice | category , action , id , Part , Value , Attribute , Price , Quantity |
| Parts | category , action , id , partNumber , Filter , Team , Quantity , Pricing |
| Employee\_Details | category , action , id , Name , PhoneNumber , DateofBirth , Salary |
| CN\_Credentials | category , action , id , integrationName , username , password , desc |
| PetPrices | category , action , id , Type , PartNumber , PetPrice |
| TestTable | category , action , id , Column1 |
| Oracle\_BomItemDef | category , action , id , VariableName , ItemId , Name , ItemType , PartNumber , Optional , SalesItem , ParentVariableName , RootVariableName , EffectiveFrom , EffectiveTo , ManufacturingItem , IncludedInBasePrice , SequenceNum , DefaultQuantity |
| Oracle\_BomItemMap | category , action , id , VariableName , BomItemVarName , ConfigAttrVarName , ConfigAttrValue , ParentBomMapVarName , EffectiveFrom , EffectiveTo , ConfigAttrOptionVal |
| POC\_Parts | category , action , id , Opportunity , Part\_Num |
| Vertiv\_InternalParts | category , action , id , BOMItemName , ParentBOMItemName , Attr1 , Op1 , Val1 , Attr2 , Op2 , Val2 , Qty |
| Contract\_IDS | category , action , id , Contract\_ID , Contract\_Name , Oppty\_ID |
| Oracle\_aboPart2Model | category , action , id , PartNumber , ProductLine , Segment , Model |
| Vertiv\_Documents | category , action , id , ProductID , DocumentName , DocumentPath |

**Users List**

|  |  |
| --- | --- |
| User ID | User Login Name |
| 3 | superuser |
| 7 | superuser |
| 250519 | guest |
| 4119220 | superuser |
| 4119231 | guest |
| 36178341 | guest |
| 36257935 | Admin |
| 36257929 | inga.lang |
| 36279101 | mcadambi |
| 36271184 | laurent.dumont |
| 36394889 | chart |
| 36295001 | mandrews |
| 36295419 | demoadmin |
| 36379801 | luly |
| 36393264 | tferrie |
| 36295901 | hmak |
| 36295106 | tferrie2 |
| 36295111 | jbrinkman |
| 36298237 | cdetrych |
| 36302902 | lisa.jones |
| 36302908 | mateo.lopez |
| 36376201 | ggingerich |
| 36388213 | bpetrik |
| 36393846 | lsturz |
| 36442119 | ri.yang |
| 36442125 | sam.brown |
| 36442131 | takahiro.tayama |
| 36443102 | taro.hiraiwa |
| 36443120 | wei.wang |
| 36462029 | ejacobson |
| 36462035 | mmorman |
| 36403801 | mweinberg |
| 36442093 | liang.yang |
| 36442099 | lucy.harris |
| 36442106 | oliver.brunke |
| 36442112 | paul.dubois |
| 36488102 | owen.mann |
| 36481028 | lche |
| 36488076 | superuser |
| 36488082 | guest |
| 36497302 | provision |
| 36499241 | system\_user\_cpq\_server\_as\_client |
| 36547572 | colby.morris |
| 36554841 | david.wilson |
| 36742701 | ajledoux |
| 36948236 | dchang |
| 36995791 | yjiao |
| 36982277 | guest1 |
| 36997649 | stwinam |
| 37028560 | sraykar |
| 37028937 | gpulipati |
| 37163204 | rwest |
| 37163210 | sbahukhandi |
| 37163216 | speddireddy |
| 37163235 | spandhari |
| 37163240 | anandssingh |
| 37317499 | icsuser |
| 37320886 | snloya |
| 37320892 | amakuluri |
| 37321258 | sguo |
| 37029525 | julian.henderson |
| 37163228 | abinash |
| 37163247 | abkazi |
| 37163252 | mrinalk |
| 37163257 | rsharmas |
| 38441959 | bas |
| 37163262 | fosinubi |
| 37163267 | begeorge |
| 37163272 | shruti |
| 37163277 | NGupta |
| 37317207 | pagupta |
| 37336587 | rkunal |
| 37380763 | Prashant |
| 37327988 | subhdas |
| 37327998 | vepotti |
| 37324030 | vbelawade |
| 37327993 | jshaik |
| 37331965 | hrajupet |
| 37340252 | mkasturiwale |
| 37340259 | bracha |
| 37324138 | jswaminathan |
| 37324143 | bbhushan |
| 37321291 | hlee |
| 38442935 | rajakolli |
| 40770623 | rprabhakar |
| 41475905 | vsoni |
| 41586899 | Adwiti |
| 37422061 | pkrishnan |
| 37443657 | rfaigao |
| 37443662 | jbade |
| 38442892 | priyatam |
| 38442912 | ramyap |
| 38442969 | Nandita |
| 38442989 | Pallavi |
| 38882822 | kkim |
| 40514945 | traininguser |
| 40706542 | veramaswamy |
| 37386772 | adfGen |
| 40121332 | ogonzalez |
| 40688319 | plohani |
| 40706552 | bgong |
| 40779994 | Somya |
| 37387938 | ldesmet |
| 37494165 | sayanka |
| 37504721 | Test1 |
| 37504726 | Test2 |
| 37509124 | abc1 |
| 37509129 | mno1 |
| 37514283 | User2 |
| 37514288 | User3 |
| 37538971 | aaganesan |
| 37918267 | ctiwari |
| 38138007 | bflechsig |
| 38238630 | pvajjhala |
| 37388161 | pirathi |
| 37642896 | ssharma |
| 38280727 | spathade |
| 38659694 | ABhoi |
| 38714296 | rjadhwani |
| 38714306 | brjohnson |
| 40118527 | lsoto |
| 40350280 | ajain |
| 40752129 | Sirisha |
| 40975107 | aalibhai@deloitte.com |
| 41099875 | Shiwam |
| 41362736 | Teja |
| 40749551 | sradovic |

**Groups List**

|  |  |
| --- | --- |
| Group Label | Group Name |
| Sales User | salesUser |
| Business Administrator | businssAdminstrator |
| Technical Administrator | technicalAdminstrator |
| Manager User | managerUser |
| VP Users | vPUsers |
| Tablets | tablets |
| Email Group | emailGroup |
| Data Manage | dataManagement |
| Federal Users | federalUsers |
| Alta UI Users | altaUIUsers |
| Green Dots Auto Parts | greenDotsAutoParts |
| Order Management | orderManagement |
| Executive Sales Manager | execSalesMgr |
| Vertiv UI Users | vertivUIUsers |