var importNames = JavaImporter();

importNames.importPackage(Packages.java.util);

importNames.importPackage(Packages.java.lang);

importNames.importPackage(Packages.java.sql);

importNames.importPackage(Packages.br.com.centralit.citcorpore.negocio);

importNames.importPackage(Packages.br.com.centralit.citcorpore.integracao);

importNames.importPackage(Packages.br.com.centralit.citcorpore.bean);

importNames.importPackage(Packages.br.com.citframework.util);

importNames.importPackage(Packages.br.com.citframework.comparacao);

importNames.importPackage(Packages.br.com.citframework.integracao);

importNames.importPackage(Packages.com.google.gson);

importNames.importPackage(Packages.br.com.centralit.bpm.integracao);

importNames.importPackage(Packages.br.com.centralit.bpm.dto);

importNames.importPackage(Packages.br.com.centralit.citcorpore.bpm.negocio);

importNames.importPackage(Packages.br.com.citframework.excecao);

var print = java.lang.System.out;

montarSqlSolicitacoes = function() {

var sql = new importNames.StringBuilder();

sql.append("select ")

.append("distinct ")

.append("ss.idsolicitacaoservico,")

.append("ss.datahorasolicitacao,")

.append("EXTRACT(DAY FROM ss.datahorasolicitacao) diaabertura,")

.append("EXTRACT(MONTH FROM ss.datahorasolicitacao) mesabertura,")

.append("EXTRACT(YEAR FROM ss.datahorasolicitacao) anoabertura,")

.append("ss.datahorafim,")

.append("EXTRACT(DAY FROM ss.datahorafim) diafechamento,")

.append("EXTRACT(MONTH FROM ss.datahorafim) mesfechamento,")

.append("EXTRACT(YEAR FROM ss.datahorafim) anofechamento,")

.append("g.idgrupo idgrupoatual,")

.append("g.sigla grupoatual,")

.append("ss.idprioridade,")

.append("pr.nomeprioridade,")

.append("ss.idservicocontrato,")

.append("sc.idcontrato,")

.append("c.numero numerocontrato,")

.append("s.idtiposervico,")

.append("ts.nometiposervico,")

.append("s.idportfolioservico,")

.append("ps.descportfolioservico,")

.append("ss.idsolicitante,")

.append("empsol.nome solicitante,")

.append("ss.idusuarioresponsavelatual,")

.append("usurespatual.nome tecnicoresponsavel,")

.append("ss.idtipodemandaservico,")

.append("tds.classificacao tiposolicitacao,")

.append("ss.idcausaincidente,")

.append("descricaocausa causa,")

.append("ss.idcategoriasolucao,")

.append("categoriasolucao.descricaocategoriasolucao categoriasolucao,")

.append("ss.idstatus,")

.append("case ss.idstatus when 1 then 'IN\_PROGRESS' when 2 then 'SUSPENDED' when 3 then 'CANCELED' when 4 then 'SOLVED' when 5 then 'REOPENED' when 6 then 'CLOSED' when 7 then 'RECLASSIFIED' end as status,")

.append("ss.idacordonivelservico,")

.append("ss.prazohh prazosla\_hh,")

.append("ss.prazomm prazosla\_mm,")

.append("ss.idcalendario,")

.append("calendario.descricao calendario,")

.append("ss.datahoralimite,")

.append("EXTRACT(DAY FROM ss.datahoralimite) dialimitesla,")

.append("EXTRACT(MONTH FROM ss.datahoralimite) meslimitesla,")

.append("EXTRACT(YEAR FROM ss.datahoralimite) anolimitesla,")

.append("elfluxo.nome tarefaatual,")

.append("c.idcliente,")

.append("clientes.nomefantasia cliente,")

.append("c.idfornecedor,")

.append("fornecedor.nomefantasia fornecedor,")

.append("s.idcategoriaservico,")

.append("categoriaservico.nomecategoriaservico categoriaservico,")

.append("condicaooperacao.idcondicaooperacao,")

.append("condicaooperacao.nomecondicaooperacao,")

.append("ss.idorigem,")

.append("origemocorrencia.nome origemdasolicitacao,")

.append("c.idmoeda,")

.append("moedas.nomemoeda moeda,")

.append("bpm\_tipofluxo.idtipofluxo,")

.append("bpm\_tipofluxo.nomefluxo fluxo,")

.append("s.idimportancianegocio,")

.append("importancianegocio.nomeimportancianegocio,")

.append("csol.idlocalidade,")

.append("loc.nomelocalidade localidade,")

.append("ss.idunidade,")

.append("unidade.nome,")

.append("ss.urgencia,")

.append("ss.impacto,")

.append("f\_sla\_atrasado(ss.idstatus,ss.situacaosla,ss.datahoralimite,ss.datahorasuspensaosla,ss.prazohh,ss.prazomm,ss.tempoatendimentohh,ss.tempoatendimentomm) rupturasla,")

.append("ss.seqreabertura,")

.append("(select case when count(0)>0 then 'S' else 'N' end from ocorrenciasolicitacao o where o.idsolicitacaoservico = ss.idsolicitacaoservico and o.idoccurrencecategory = 13) houvereclassificacao,")

.append("tempoatendimentohh,")

.append("tempoatendimentomm,")

.append("tempoatrasohh,")

.append("tempoatrasomm,")

.append("(case when upper(s.incidentecritico) = 'S' then 'S' else 'N' end) major,")

.append("(select nota from pesquisasatisfacao pqs where rownum < 2 and pqs.idsolicitacaoservico = ss.idsolicitacaoservico) notapesquisasatisfacao,")

.append("(select count(ssqtf.idsolicitacaoservico) from solicitacaoservico ssqtf where ssqtf.idsolicitacaorelacionada = ss.idsolicitacaoservico and ssqtf.idstatus<>3) qtdesolicitacoesfilhas,")

.append("(select count(ssqtsb.idsolicitacaoservico) from solicitacaoservico ssqtsb where ssqtsb.idsolicitacaopai = ss.idsolicitacaoservico and ssqtsb.idstatus<>3) qtdesubsolicitacoes,")

.append("(select count(chss.idbaseconhecimento) from conhecimentosolicitacaoservico chss where chss.idsolicitacaoservico = ss.idsolicitacaoservico) qtdebaseconhecimento,")

.append("(select count(sspr.idproblema) from solicitacaoservicoproblema sspr join problema pr on sspr.idproblema = pr.idproblema where sspr.idsolicitacaoservico = ss.idsolicitacaoservico and pr.idstatus<>8) qtdeproblemas,")

.append("(select count(libss.idliberacao) from liberacaosolicitacaoservico libss join liberacao lib on libss.idliberacao = lib.idliberacao where libss.idsolicitacaoservico = ss.idsolicitacaoservico and lib.idstatus<>8) qtdeliberacao,")

.append("(select count(ssmud.idrequisicaomudanca) from solicitacaoservicomudanca ssmud join requisicaomudanca mud on ssmud.idrequisicaomudanca = mud.idrequisicaomudanca where ssmud.idsolicitacaoservico = ss.idsolicitacaoservico and mud.idstatus<>8) qtdemudancas,")

.append("(select count(icfgss.iditemconfiguracao) from itemcfgsolicitacaoserv icfgss join itemconfiguracao itcfg on icfgss.iditemconfiguracao = itcfg.iditemconfiguracao where icfgss.idsolicitacaoservico = ss.idsolicitacaoservico and (icfgss.datafim is null) and (itcfg.datafim is null)) qtdeics,")

.append("(select count(distinct(apliserv.idaplicacao)) ")

.append("from servicoautorelacionamento sautorel join servico serv on sautorel.idservico = serv.idservico ")

.append("join aplicacaoservico apliserv on sautorel.idservicorelacionado = apliserv.idservico ")

.append("join aplicacao on apliserv.idaplicacao = aplicacao.idaplicacao ")

.append("where serv.idservico=sc.idservico and (upper(serv.deleted) <> 'Y') and (sautorel.datafim is null) and (aplicacao.datafim is null)) qtdeaplicacoes,")

.append("(select count(prj.idprojeto) from projetos2 prj where prj.idprojeto = ss.idprojeto and (prj.datafim is null)) qtdeprojetos,")

.append("(select count(idcontroleged) from controleged where idtabela=3 and id = ss.idsolicitacaoservico) qtdeanexos,")

.append("(select count(distinct(ativper.idatividadeperiodica)) from atividadeperiodica ativper where ativper.idsolicitacaoservico = ss.idsolicitacaoservico and (ativper.datafim is null)) qtdeagendamentoatividades,")

.append("(select count(ap.idatividadeperiodica) ")

.append("from atividadeperiodica ap join execucaoatividadeperiodica exap on ap.idatividadeperiodica = exap.idatividadeperiodica ")

.append("where ap.idsolicitacaoservico = ss.idsolicitacaoservico and (ap.datafim is null) and (upper(exap.situacao) = 'F') ")

.append(") qtdeagendamentativfinalizadas,")

.append("case when upper(c.tipo) = 'U' then 'S' else 'N' end contratoapoio,")

.append("case when upper(tiposervico) = 'A' then 'S' else 'N' end servicoapoio,")

.append("ss.custoservico,")

.append("case when upper(tds.classificacao) = 'I' then 'S' else 'N' end servicoindisponivel,")

.append("(select count(idpesquisasatisfacao) ")

.append("from pesquisasatisfacao psatis join grausatisfacao graus on psatis.nota= graus.idgrausatisfacao ")

.append("where psatis.idsolicitacaoservico = ss.idsolicitacaoservico and notaavaliacao in (3,4)) qtdeelogios,")

.append("(select count(idpesquisasatisfacao) ")

.append("from pesquisasatisfacao psatis join grausatisfacao graus on psatis.nota= graus.idgrausatisfacao ")

.append("where psatis.idsolicitacaoservico = ss.idsolicitacaoservico and notaavaliacao in (0,1)) qtdequeixas,")

.append("(case when upper(tds.classificacao)='P' then 'S' else 'N' end) procedimentocontinuidade,")

.append("sc.custohoraindisponibilidade custoindisponibilidade,")

.append("(select sar.idservicorelacionado from servicoautorelacionamento sar where rownum = 1 and sar.idservico = s.idservico) as idservico,")

.append("(select srv.nomeservico from servico srv where srv.idservico in (select sar.idservicorelacionado from servicoautorelacionamento sar where rownum = 1 and sar.idservico = s.idservico)) as nomeservico,")

.append("s.idservico idatividade, s.nomeservico nomeatividade ")

.append("from solicitacaoservico ss join table(f\_execucao\_solicitacao(null,null,null,null)) f on ss.idsolicitacaoservico = f.idsolicitacaoservico ")

.append("left join bpm\_atribuicaofluxo a on f.iditemtrabalho = a.iditemtrabalho ")

.append("left join grupo g on a.idgrupo = g.idgrupo ")

.append("left join prioridade pr on ss.idprioridade = pr.idprioridade ")

.append("left join servicocontrato sc on ss.idservicocontrato = sc.idservicocontrato ")

.append("left join contratos c on sc.idcontrato = c.idcontrato ")

.append("left join servico s on sc.idservico = s.idservico ")

.append("left join tiposervico ts on s.idtiposervico = ts.idtiposervico ")

.append("left join portfolioservico ps on s.idportfolioservico = ps.idportfolioservico ")

.append("left join empregados empsol on ss.idsolicitante = empsol.idempregado ")

.append("left join bpm\_itemtrabalhofluxo i on f.iditemtrabalho = i.iditemtrabalho ")

.append("left join usuario usurespatual on i.idresponsavelatual = usurespatual.idusuario ")

.append("left join tipodemandaservico tds on ss.idtipodemandaservico = tds.idtipodemandaservico ")

.append("left join causaincidente on ss.idcausaincidente = causaincidente.idcausaincidente ")

.append("left join categoriasolucao on ss.idcategoriasolucao = categoriasolucao.idcategoriasolucao ")

.append("left join calendario on ss.idcalendario = calendario.idcalendario ")

.append("left join bpm\_elementofluxo elfluxo on i.idelemento = elfluxo.idelemento ")

.append("left join clientes on c.idcliente = clientes.idcliente ")

.append("left join fornecedor on c.idfornecedor = fornecedor.idfornecedor ")

.append("left join categoriaservico on s.idcategoriaservico = categoriaservico.idcategoriaservico ")

.append("left join condicaooperacao on sc.idcondicaooperacao = condicaooperacao.idcondicaooperacao ")

.append("left join origemocorrencia on origemocorrencia.idorigemocorrencia = ss.idorigem ")

.append("left join moedas on c.idmoeda = moedas.idmoeda ")

.append("left join execucaosolicitacao e on ss.idsolicitacaoservico = e.idsolicitacaoservico ")

.append("left join bpm\_fluxo on e.idfluxo = bpm\_fluxo.idfluxo ")

.append("left join bpm\_tipofluxo on bpm\_fluxo.idtipofluxo = bpm\_tipofluxo.idtipofluxo ")

.append("left join importancianegocio on s.idimportancianegocio = importancianegocio.idimportancianegocio ")

.append("left join contatosolicitacaoservico csol on ss.idcontatosolicitacaoservico = csol.idcontatosolicitacaoservico ")

.append("left join localidade loc on csol.idlocalidade = loc.idlocalidade ")

.append("left join unidade on ss.idunidade = unidade.idunidade ")

.append("where dtlastmodification > TO\_TIMESTAMP(TO\_CHAR(CURRENT\_TIMESTAMP - interval '1' day, 'YYYY-MM-DD'), 'YYYY-MM-DD HH24:MI:SS') and a.idtype=1 ")

.append("order by ss.idsolicitacaoservico desc");

return sql.toString();

}

executaConsultaSolicitacoes = function(){

return jdbcEngine.execSQL(montarSqlSolicitacoes(), null, 0);

}

montarSqlSolicitacaoCadastrada = function(){

var sql = new importNames.StringBuilder();

sql.append("select idsolicitacaoservico from fato\_solicitacaoservico where idsolicitacaoservico=?");

return sql.toString();

}

solicitacaoCadastrada = function(idSolicitacaoServico){

var cadastrada = false;

var parmsUtilizadosNoSQL = new importNames.ArrayList();

parmsUtilizadosNoSQL.add(idSolicitacaoServico);

var consulta = jdbcEngine.execSQL(montarSqlSolicitacaoCadastrada(), parmsUtilizadosNoSQL.toArray(), 0);

if((consulta!=null)&&(consulta.size()>0)){

var registroSolicitacao = null;

for (var itSolicitacao = consulta.iterator(); itSolicitacao.hasNext();){

registroSolicitacao = itSolicitacao.next();

if (registroSolicitacao[0]!=null && registroSolicitacao[0]>0){

cadastrada = true;

}

}

}

return cadastrada;

}

montarSqlInserirDados = function(){

var sql = new importNames.StringBuilder();

sql.append("INSERT INTO fato\_solicitacaoservico(")

.append("idsolicitacaoservico, datahorasolicitacao, diaabertura, mesabertura,")

.append("anoabertura, datahorafim, diafechamento, mesfechamento, anofechamento,")

.append("idgrupoatual, grupoatual, idprioridade, nomeprioridade, idservicocontrato,")

.append("idcontrato, numerocontrato, idtiposervico, nometiposervico, idportfolioservico,")

.append("descportfolioservico, idsolicitante, solicitante, idusuarioresponsavelatual,")

.append("tecnicoresponsavel, idtipodemandaservico, tiposolicitacao, idcausaincidente,")

.append("causa, idcategoriasolucao, categoriasolucao, idstatus, status,")

.append("idacordonivelservico, prazosla\_hh, prazosla\_mm, idcalendario,")

.append("calendario, datahoralimite, dialimitesla, meslimitesla, anolimitesla,")

.append("tarefaatual, idcliente, cliente, idfornecedor, fornecedor, idcategoriaservico,")

.append("categoriaservico, idcondicaooperacao, nomecondicaooperacao, idorigem,")

.append("origemdasolicitacao, idmoeda, moeda, idtipofluxo, fluxo, idimportancianegocio,")

.append("importanciaservicoaonegocio, idlocalidade, localidade, idunidade,")

.append("unidade, urgencia, impacto, rupturasla, qtdereaberturas, houvereclassificacao,")

.append("tempoatendimentohh, tempoatendimentomm, tempoatrasohh, tempoatrasomm,")

.append("major, notapesquisasatisfacao, qtdesolicitacoesfilhas, qtdesubsolicitacoes,")

.append("qtdebaseconhecimento, qtdeproblemas, qtdeliberacao, qtdemudancas,")

.append("qtdeics, qtdeaplicacoes, qtdeprojetos, qtdeanexos, qtdeagendamentoatividades,")

.append("qtdeagendamentativfinalizadas, contratoapoio, servicoapoio, custoservico,")

.append("servicoindisponivel, qtdeelogios, qtdequeixas, procedimentocontinuidade,")

.append("custoindisponibilidade,idservico,nomeservico,idatividade,nomeatividade) VALUES (")

.append("?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,")

.append("?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?")

.append(")");

return sql.toString();

}

inserirDados = function(dados){

try {

jdbcEngine.execUpdate(montarSqlInserirDados(), dados);

} catch(err) {

print.println("Problema ao inserir dados na tabela Fato de Solicitacao - "+dados[0]);

print.println(err);

}

}

montarSqlAtualizarDados = function(){

var sql = new importNames.StringBuilder();

sql.append("UPDATE fato\_solicitacaoservico SET ")

.append("datahorasolicitacao=?,diaabertura=?,")

.append("mesabertura=?,anoabertura=?,datahorafim=?,diafechamento=?,")

.append("mesfechamento=?,anofechamento=?,idgrupoatual=?,grupoatual=?,")

.append("idprioridade=?,nomeprioridade=?,idservicocontrato=?,idcontrato=?,")

.append("numerocontrato=?,idtiposervico=?,nometiposervico=?,idportfolioservico=?,")

.append("descportfolioservico=?,idsolicitante=?,solicitante=?,idusuarioresponsavelatual=?,")

.append("tecnicoresponsavel=?,idtipodemandaservico=?,tiposolicitacao=?,")

.append("idcausaincidente=?,causa=?,idcategoriasolucao=?,categoriasolucao=?,")

.append("idstatus=?,status=?,idacordonivelservico=?,prazosla\_hh=?,")

.append("prazosla\_mm=?,idcalendario=?,calendario=?,datahoralimite=?,")

.append("dialimitesla=?,meslimitesla=?,anolimitesla=?,tarefaatual=?,")

.append("idcliente=?,cliente=?,idfornecedor=?,fornecedor=?,idcategoriaservico=?,")

.append("categoriaservico=?,idcondicaooperacao=?,nomecondicaooperacao=?,")

.append("idorigem=?,origemdasolicitacao=?,idmoeda=?,moeda=?,idtipofluxo=?,")

.append("fluxo=?,idimportancianegocio=?,importanciaservicoaonegocio=?,")

.append("idlocalidade=?,localidade=?,idunidade=?,unidade=?,urgencia=?,")

.append("impacto=?,rupturasla=?,qtdereaberturas=?,houvereclassificacao=?,")

.append("tempoatendimentohh=?,tempoatendimentomm=?,tempoatrasohh=?,")

.append("tempoatrasomm=?,major=?,notapesquisasatisfacao=?,qtdesolicitacoesfilhas=?,")

.append("qtdesubsolicitacoes=?,qtdebaseconhecimento=?,qtdeproblemas=?,")

.append("qtdeliberacao=?,qtdemudancas=?,qtdeics=?,qtdeaplicacoes=?,")

.append("qtdeprojetos=?,qtdeanexos=?,qtdeagendamentoatividades=?,qtdeagendamentativfinalizadas=?,")

.append("contratoapoio=?,servicoapoio=?,custoservico=?,servicoindisponivel=?,")

.append("qtdeelogios=?,qtdequeixas=?,procedimentocontinuidade=?,custoindisponibilidade=?,")

.append("idservico=?,nomeservico=?,idatividade=?,nomeatividade=?,")

.append("datahoracarga=CURRENT\_TIMESTAMP ")

.append("WHERE idsolicitacaoservico = ?");

return sql.toString();

}

atualizarDados = function(dados){

try {

var parmsUtilizadosNoSQL = new importNames.ArrayList();

if(dados!=null){

for (var i = 1; i < dados.length; i++) {

parmsUtilizadosNoSQL.add(dados[i]);

}

}

parmsUtilizadosNoSQL.add(dados[0]);

jdbcEngine.execUpdate(montarSqlAtualizarDados(), parmsUtilizadosNoSQL.toArray());

} catch(err) {

print.println("Problema ao atualizar dados na tabela Fato de Solicitacao - "+dados[0]);

}

}

gravarDados = function(dados){

if (solicitacaoCadastrada(dados[0])){

atualizarDados(dados);

} else {

inserirDados(dados);

}

}

gerarCargaTabelaFatoSolicitacao = function(){

var consulta = executaConsultaSolicitacoes();

var registroSolicitacao = null;

print.println("Inicio Carga na Tabela Fato Solicitacao de Servico:");

if((consulta!=null)&&(consulta.size()>0)){

for (var itSolicitacao = consulta.iterator(); itSolicitacao.hasNext();){

registroSolicitacao = itSolicitacao.next();

gravarDados(registroSolicitacao);

print.println("Solicitacao: "+registroSolicitacao[0]);

}

}

return "Fim - Carga na Tabela Fato Solicitacao de Servico";

}

" "+gerarCargaTabelaFatoSolicitacao();