



pears.NetClientVacanciesLineStatusUpdate



Generated schema reference. Regenerate this page from the SQL unload; keep hand-maintained business notes in the narrative namespace.

Original SQL

```
CREATE PROCEDURE "pears"."NetClientVacanciesLineStatusUpdate"( IN
  "pWebUserID" CHAR(20),IN "pLineNo" CHAR(21),IN "pTransactionType" CHAR(1),IN
  "pNote" CHAR(255),IN "pOurRef" CHAR(20),IN "pCancelReason" CHAR(1) DEFAULT
  NULL,IN "pLocation" CHAR(100) DEFAULT NULL,IN "pPersonID" CHAR(20) DEFAULT
  NULL )
RESULT( "retval" CHAR(250) )
// IQXWeb
BEGIN
  DECLARE "PorT" CHAR(1);
  DECLARE "LineNo" CHAR(20);
  DECLARE "VacID" CHAR(20);
  DECLARE "StaffID" CHAR(20);
  DECLARE "LShiftDate" DATE;
  DECLARE "LTimeFrom" TIME;
  DECLARE "LTimeTo" TIME;
  DECLARE "sid" CHAR(20);
  DECLARE "LTempShiftPlanID" CHAR(20);
  DECLARE "LWhoCancelled" CHAR(60);
  DECLARE "RequiresConfirm" INTEGER;
  DECLARE "MustTempConfirm" INTEGER;
  DECLARE "Note" CHAR(255);
  DECLARE "IntNote" CHAR(255);
  DECLARE "ShiftRef" CHAR(20);
  DECLARE "Location" CHAR(100);
  DECLARE "EmploymentID" CHAR(20);
  DECLARE "PersID" CHAR(20);
  DECLARE "IntermediateString" CHAR(150);
  SET "PorT" = "left"("pLineNo",1); -- discover whether TempShiftPlan or
TempShift
  SET "LineNo" = "right"("pLineNo","length"("pLineNo")-1);
  CASE "PorT"
  WHEN 'V' THEN
    (SELECT
      "t"."vacancyid","shiftdate","timefrom","timeto","tempshiftplanid","e"."emplo
ymentid",NULL,"v"."refcode" INTO
```



```
"VacID","LShiftDate","LTimeFrom","LTimeTo","LTempShiftPlanID","EmploymentID"
,"PersID","ShiftRef" FROM "tempshiftplan" AS "t" KEY JOIN "vacancy" AS "v"
KEY JOIN "employment" AS "e" WHERE "v"."vacancyid" = "LineNo")
  WHEN 'P' THEN
    (SELECT
      "t"."vacancyid","t"."shiftdate","t"."timefrom","t"."timeto","t"."tempshiftpl
      anid","e"."employmentid","p"."personid","tsp"."shiftserialnumber" INTO
      "VacID","LShiftDate","LTimeFrom","LTimeTo","LTempShiftPlanID","EmploymentID"
      ,"PersID","ShiftRef" FROM "tempshiftplan" AS "tsp" KEY JOIN "tempshift" AS
      "t" KEY JOIN "vacancy" KEY JOIN "employment" AS "e" LEFT OUTER JOIN "person"
      AS "p" ON "p"."personid" = "t"."personid" WHERE "tsp"."TempShiftPlanID" =
      "LineNo")
    ELSE
      RETURN
    END CASE;
  SET "PersID" = "isnull"("PersID","pPersonID");
  -- conFirm
  IF "pTransactionType" = 'F' THEN
    CALL
    "IQXNetPopup"("pWebUserID","PersID","EmploymentID","VacID",NULL,NULL,'SHIFTS
    TATECHANGE','Shift Confirmed',"pNote");
    UPDATE "tempshift" SET "clientconfirmed" = 1 WHERE "tempshiftplanid" =
    "LineNo";
    (SELECT "t"."tempconfirmed","p"."tempmustconfirm" INTO
    "RequiresConfirm","MustTempConfirm" FROM "tempshift" AS "t" KEY JOIN
    "tempshiftplan" AS "p" WHERE "p"."tempshiftplanid" = "LineNo");
    IF "RequiresConfirm" = 0 AND "MustTempConfirm" = 1 THEN SELECT
    '1:~success#PProvisional:requires confirmation by temp'
    ELSE SELECT '1:~success#BConfirmed:awaits timesheet'
    END IF END IF;
  -- caNcel or Reject
  IF "pTransactionType" IN( 'N','R' ) THEN
    SET "StaffID" = (SELECT "staffid" FROM "vacancy" WHERE "vacancyid" =
    "VacID");
    SET "LWhoCancelled" = (SELECT "Staffid" FROM "iqxnetuser" WHERE
    "iqxnetuserid" = "pWebUserID");
    --set UWhoCancelled='Cancelled by '+isnull((select name from iqxnetuser
    where iqxnetuserid = pWebUserID),'not recorded');
    SET "PersID" = (SELECT "personid" FROM "tempshift" RIGHT OUTER JOIN
    "tempshiftplan" AS "p" WHERE "p"."tempshiftplanid" = "LineNo");
    UPDATE "tempshift" SET "state" = 'C',"cancelreason" =
    "pCancelReason","whencancelled" = CURRENT_TIMESTAMP,"confirmedwith" =
    "LWhoCancelled","WhoCancelled" = "LWhoCancelled" WHERE "tempshiftplanid" =
    "LineNo";
    IF @@Rowcount = 0 THEN
      SET "sid" = "uniquekey"('1');
      INSERT INTO "tempshift"(
      "tempshiftid","vacancyid","shiftdate","timefrom","timeto","state","tempshift
```



```
planid","cancelreason","whencancelled","confirmedwith" ) VALUES(
"sid","VacID","LShiftDate","LTimeFrom","LTimeTo",'C',"LTempShiftPlanID","pCa
ncelReason",CURRENT_TIMESTAMP,"LWhoCancelled" )
    END IF;
    IF "pTransactionType" = 'R' THEN
        CALL
"IXQNetPopup"("pWebUserID","pPersonID","EmploymentID","VacID",NULL,NULL,'SHI
FTSTATECHANGE','Shift Candidate Rejected - '+'ShiftRef',"pNote")
    ELSE
        CALL
"IXQNetPopup"("pWebUserID","pPersonID","EmploymentID","VacID",NULL,NULL,'SHI
FTSTATECHANGE','Shift Cancelled - '+'ShiftRef',"pNote")
    END IF;
    SELECT '1:~success#CCancelled:'+(SELECT "name" FROM
"TempShiftCancelReason" WHERE "TempShiftCancelReasonID" = "pCancelReason")
    END IF;
    -- reQuest
    IF "pTransactionType" = 'Q' THEN
        SET "Note" =
"substring"("pNote","charindex('Request:', "pNote")+9,"charindex(';', "pNote
")-10);
        CALL
"IXQNetPopup"("pWebUserID","PersID","EmploymentID","VacID",NULL,NULL,'SHIFTS
TATECHANGE','Requested candidate - '+'ShiftRef',"pNote");
        SELECT '0:~success'
    END IF;
    -- coMment
    IF "pTransactionType" = 'M' THEN
        SET "Note" = "substring"("pNote","charindex('Note:', "pNote")+6);
        SET "Note" = "substring"("Note",1,149);
        SET "IntermediateString" = "substring"("pNote","charindex('Shift
Ref', "pNote")+10);
        SET "ShiftRef" =
"substring"("IntermediateString",1,"charindex('Note:', "IntermediateString")
-3);
        CALL
"IXQNetPopup"("pWebUserID","PersID","EmploymentID","VacID",NULL,NULL,'SHIFTS
TATECHANGE','Shift Note Added - '+'ShiftRef',"pNote");
        SELECT '0:~success'
    END IF;
    -- Location
    IF "pTransactionType" = 'L' THEN
        SET "Note" =
"substring"("pNote","charindex('Note:', "pNote")+9,"charindex(';', "pNote")-
10);
        SET "IntermediateString" = "substring"("pNote","charindex('Shift
Ref', "pNote")+10);
        SET "ShiftRef" =
```



```
"substring"("IntermediateString",1,"charindex"('Note:',"IntermediateString")
-3);
UPDATE "tempshiftplan" SET "ClientNote" = "pLocation" WHERE
"TempShiftPlanID" = "LTempShiftPlanID";
CALL
"IQXNetPopup"("pWebUserID","PersID","EmploymentID","VacID",NULL,NULL,'SHIFTS
TATECHANGE','Shift Location modified - '+'ShiftRef',"pNote"+'Location:
'+"pLocation");
SELECT '0:~success'
END IF
END /* DOC
```

2016-11-29 PC doc and test
only using Request and Reject

F=Confirm shift (not used at the moment)
R=Reject candidate
N=Cancel shift (not used at the moment)
Q=Request candidate
M=Shift Note (not used at the moment)
L=Change location of shift (not used at the moment)

Tests:

all invalid

valid pWebUserID

valid pWebUserID,valid pLineNo

valid pWebUserID,valid pLineNo, invalid pTransactionType

*valid pWebUserID,valid pLineNo, pTransactionType=Q, pNote, pPersonID for
vacancy*

valid pWebUserID,valid pLineNo, pTransactionType=R, pNote, pPersonID

*valid pWebUserID,valid pLineNo, pTransactionType=Q, pNote, pPersonID for
Tempshift*

**/*

/ TEST*

call NetTestSetup('');

*update iqxnetuserlink set employmentid=null where IQXNetUserLinkID in
('test.client','test.agency','test.candidate','test.owner');*

*update iqxnetuserlink set employmentid='TEST' where
IQXNetUserLinkID='test.client';*

*select * from*

*NetClientVacanciesLineStatusUpdate(null,null,null,null,null,null,null);
expect EOF;*

*select * from*

*NetClientVacanciesLineStatusUpdate('test.client',null,null,null,null,null,nu
ll,null);*

expect EOF;

*select * from*

*NetClientVacanciesLineStatusUpdate('test.client','VTEST',null,null,null,
null,null);*



```
expect EOF;
select * from
NetClientVacanciesLineStatusUpdate('test.client','VTEST','Q','TestNote',null
,null,null,'test');
expect 0*;
select top 1 * from diary where description like 'Requested Candidate%'
order by whenentered desc;
expect
staffid=TEST,personid=test,vacancyID=TEST,EmploymentID=TEST,notes=TestNote;
select * from
NetClientVacanciesLineStatusUpdate('test.client','VTEST','R','TestNote',null
,null,null,'test');
expect retval=1:~success#CCancelled;;
select top 1 * from diary where description like 'Shift Candidate Rejected%'
order by whenentered desc;
expect
staffid=TEST,personid=test,vacancyID=TEST,EmploymentID=TEST,notes=TestNote;
select * from
NetClientVacanciesLineStatusUpdate('test.client','PTEST','Q','TestNote',null
,null,null,'test');
expect 0*;
select top 1 * from diary where description like 'Requested Candidate%'
order by whenentered desc;
expect
staffid=TEST,personid=test,vacancyID=TEST,EmploymentID=TEST,notes=TestNote;
select * from
NetClientVacanciesLineStatusUpdate('test.client','PTEST','R','TestNote',null
,null,null,'test');
expect retval=1:~success#CCancelled;;
select top 1 * from diary where description like 'Shift Candidate Rejected%'
order by whenentered desc;
expect
staffid=TEST,personid=test,vacancyID=TEST,EmploymentID=TEST,notes=TestNote;
*/
GO
```

COMMENT TO PRESERVE FORMAT ON PROCEDURE

```
"pears"."NetClientVacanciesLineStatusUpdate" IS
{CREATE PROCEDURE pears."NetClientVacanciesLineStatusUpdate"( IN
"pWebUserID" CHAR(20),IN "pLineNo" CHAR(21),IN "pTransactionType" CHAR(1),IN
"pNote" CHAR(255),IN "pOurRef" CHAR(20),IN "pCancelReason" CHAR(1) DEFAULT
NULL,IN "pLocation" CHAR(100) DEFAULT NULL,IN pPersonID CHAR(20) DEFAULT
NULL )
RESULT( "retval" CHAR(250) )
// IQXWeb
BEGIN
DECLARE "PorT" CHAR(1);
DECLARE LineNo CHAR(20);
```



```
DECLARE "VacID" CHAR(20);
DECLARE "StaffID" CHAR(20);
DECLARE "LShiftDate" DATE;
DECLARE "LTimeFrom" TIME;
DECLARE "LTimeTo" TIME;
DECLARE "sid" CHAR(20);
DECLARE "LTempShiftPlanID" CHAR(20);
DECLARE "LWhoCancelled" CHAR(60);
DECLARE "RequiresConfirm" INTEGER;
DECLARE "MustTempConfirm" INTEGER;
DECLARE "Note" CHAR(255);
DECLARE "IntNote" CHAR(255);
DECLARE "ShiftRef" CHAR(20);
DECLARE "Location" CHAR(100);
DECLARE "EmploymentID" CHAR(20);
DECLARE "PersID" CHAR(20);
DECLARE "IntermediateString" CHAR(150);

SET "PorT" = "left"("pLineNo",1); -- discover whether TempShiftPlan or
TempShift
SET LineNo = RIGHT(pLineNo,LENGTH(pLineNo)-1);
CASE PorT
  WHEN 'V' THEN
    (SELECT
      "t"."vacancyid","shiftdate","timefrom","timeto","tempshiftplanid","e"."employmentid",NULL,v.refcode INTO
      "VacID","LShiftDate","LTimeFrom","LTimeTo","LTempShiftPlanID","EmploymentID"
      ,"PersID",ShiftRef FROM "tempshiftplan" AS "t" KEY JOIN "vacancy" v KEY JOIN
      "employment" AS "e" WHERE v.vacancyid = "LineNo");
    WHEN 'P' THEN
      (SELECT
        "t"."vacancyid","t"."shiftdate","t"."timefrom","t"."timeto","t"."tempshiftplanid","e"."employmentid","p"."personid",tsp.shiftserialnumber INTO
        "VacID","LShiftDate","LTimeFrom","LTimeTo","LTempShiftPlanID","EmploymentID"
        ,"PersID",ShiftRef FROM "tempshiftplan" AS "tsp" KEY JOIN "tempshift" AS "t"
        KEY JOIN "vacancy" KEY JOIN "employment" AS "e" LEFT OUTER JOIN "person" AS
        "p" ON "p"."personid" = "t"."personid" WHERE "tsp"."TempShiftPlanID" =
        "LineNo")
    ELSE
      RETURN;
  END CASE;
SET PersID = isnull(PersID,pPersonID);
-- conFirm
IF "pTransactionType" = 'F' THEN
  CALL
  "IQXNetPopup"("pWebUserID","PersID","EmploymentID","VacID",NULL,NULL,'SHIFTS
  TATECHANGE','Shift Confirmed',"pNote");
  UPDATE "tempshift" SET "clientconfirmed" = 1 WHERE "tempshiftplanid" =
```



```
"LineNo";
  (SELECT "t"."tempconfirmed","p"."tempmustconfirm" INTO
"RequiresConfirm","MustTempConfirm" FROM "tempshift" AS "t" KEY JOIN
"tempshiftplan" AS "p" WHERE "p"."tempshiftplanid" = "LineNo");
  IF "RequiresConfirm" = 0 AND "MustTempConfirm" = 1 THEN SELECT
'1::~success#PProvisional:requires confirmation by temp'
  ELSE SELECT '1::~success#BConfirmed:awaits timesheet'
  END IF END IF;
-- caNcel or Reject
IF "pTransactionType" IN( 'N','R' ) THEN
  SET "StaffID" = (SELECT "staffid" FROM "vacancy" WHERE "vacancyid" =
"VacID");
  SET "LWhoCancelled" = (SELECT "Staffid" FROM "iqxnetuser" WHERE
"iqxnetuserid" = "pWebUserID");
  --set UWhoCancelled='Cancelled by '+isnull((select name from iqxnetuser
where iqxnetuserid = pWebUserID),'not recorded');
  SET "PersID" = (SELECT "personid" FROM "tempshift" RIGHT OUTER JOIN
"tempshiftplan" p WHERE p.tempshiftplanid = "LineNo");
  UPDATE "tempshift" SET "state" = 'C',"cancelreason" =
"pCancelReason","whencancelled" = CURRENT_TIMESTAMP,"confirmedwith" =
"LWhoCancelled","WhoCancelled" = "LWhoCancelled" WHERE "tempshiftplanid" =
"LineNo";
  IF @@Rowcount = 0 THEN
    SET "sid" = "uniquekey"('1');
    INSERT INTO "tempshift"(
"tempshiftid","vacancyid","shiftdate","timefrom","timeto","state","tempshift
planid","cancelreason","whencancelled","confirmedwith" ) VALUES(
"sid","VacID","LShiftDate","LTimeFrom","LTimeTo",'C',"LTempShiftPlanID","pCa
ncelReason",CURRENT_TIMESTAMP,"LWhoCancelled" )
  END IF;
  IF "pTransactionType" = 'R' THEN
    CALL
"IQXNetPopup"("pWebUserID","pPersonID","EmploymentID","VacID",NULL,NULL,'SHI
FTSTATECHANGE','Shift Candidate Rejected - '+'ShiftRef","pNote")
  ELSE
    CALL
"IQXNetPopup"("pWebUserID","pPersonID","EmploymentID","VacID",NULL,NULL,'SHI
FTSTATECHANGE','Shift Cancelled - '+'ShiftRef","pNote")
  END IF;
  SELECT '1::~success#CCancelled:'+(SELECT "name" FROM
"TempShiftCancelReason" WHERE "TempShiftCancelReasonID" = "pCancelReason")
  END IF;
-- reQuest
IF "pTransactionType" = 'Q' THEN
  SET "Note" =
"substring"("pNote","charindex"('Request:', "pNote")+9,"charindex"(';',"pNote
")-10);
  CALL
```



```
"IQXNetPopup"("pWebUserID","PersID","EmploymentID","VacID",NULL,NULL,'SHIFTS
TATECHANGE','Requested candidate - '+'ShiftRef","pNote");
    SELECT '0:~success'
END IF;
-- coMment
IF "pTransactionType" = 'M' THEN
    SET "Note" = "substring"("pNote","charindex"('Note:','pNote')+6);
    SET "Note" = "substring"("Note",1,149);
    SET "IntermediateString" = "substring"("pNote","charindex"('Shift
Ref','pNote')+10);
    SET "ShiftRef" =
"substring"("IntermediateString",1,"charindex"('Note:','IntermediateString')
-3);
    CALL
"IQXNetPopup"("pWebUserID","PersID","EmploymentID","VacID",NULL,NULL,'SHIFTS
TATECHANGE','Shift Note Added - '+'ShiftRef","pNote");
    SELECT '0:~success'
END IF;
-- Location
IF "pTransactionType" = 'L' THEN
    SET "Note" =
"substring"("pNote","charindex"('Note:','pNote')+9,"charindex"(';','pNote')-
10);
    SET "IntermediateString" = "substring"("pNote","charindex"('Shift
Ref','pNote')+10);
    SET "ShiftRef" =
"substring"("IntermediateString",1,"charindex"('Note:','IntermediateString')
-3);
    UPDATE "tempshiftplan" SET "ClientNote" = "pLocation" WHERE
"TempShiftPlanID" = "LTempShiftPlanID";
    CALL
"IQXNetPopup"("pWebUserID","PersID","EmploymentID","VacID",NULL,NULL,'SHIFTS
TATECHANGE','Shift Location modified - '+'ShiftRef","pNote"+'Location:
'+"pLocation");
    SELECT '0:~success'
END IF
END

/* DOC
2016-11-29 PC doc and test
    only using Request and Reject

    F=Confirm shift (not used at the moment)
    R=Reject candidate
    N=Cancel shift (not used at the moment)
    Q=Request candidate
    M=Shift Note (not used at the moment)
    L=Change location of shift (not used at the moment)
```



Tests:

```
all invalid
valid pWebUserID
valid pWebUserID,valid pLineNo
valid pWebUserID,valid pLineNo, invalid pTransactionType
valid pWebUserID,valid pLineNo, pTransactionType=Q, pNote, pPersonID for
vacancy
valid pWebUserID,valid pLineNo, pTransactionType=R, pNote, pPersonID
valid pWebUserID,valid pLineNo, pTransactionType=Q, pNote, pPersonID for
Tempshift
*/
```

```
/* TEST
call NetTestSetup('');
update iqxnetuserlink set employmentid=null where IQXNetUserLinkID in
('test.client','test.agency','test.candidate','test.owner');
update iqxnetuserlink set employmentid='TEST' where
IQXNetUserLinkID='test.client';
select * from
NetClientVacanciesLineStatusUpdate(null,null,null,null,null,null,null,null);
expect EOF;
select * from
NetClientVacanciesLineStatusUpdate('test.client',null,null,null,null,null,null,null);
expect EOF;
select * from
NetClientVacanciesLineStatusUpdate('test.client','VTEST',null,null,null,null,null,null);
expect EOF;
select * from
NetClientVacanciesLineStatusUpdate('test.client','VTEST','Q','TestNote',null,null,null,'test');
expect 0*;
select top 1 * from diary where description like 'Requested Candidate%'
order by whenentered desc;
expect
staffid=TEST,personid=test,vacancyID=TEST,EmploymentID=TEST,notes=TestNote;
select * from
NetClientVacanciesLineStatusUpdate('test.client','VTEST','R','TestNote',null,null,null,'test');
expect retval=1:~success#CCancelled;;
select top 1 * from diary where description like 'Shift Candidate Rejected%'
order by whenentered desc;
expect
staffid=TEST,personid=test,vacancyID=TEST,EmploymentID=TEST,notes=TestNote;
select * from
NetClientVacanciesLineStatusUpdate('test.client','PTEST','Q','TestNote',null
```



```
,null,null,'test');  
expect 0*;  
select top 1 * from diary where description like 'Requested Candidate%'  
order by whenentered desc;  
expect  
staffid=TEST,personid=test,vacancyID=TEST,EmploymentID=TEST,notes=TestNote;  
select * from  
NetClientVacanciesLineStatusUpdate('test.client','PTEST','R','TestNote',null  
,null,null,'test');  
expect retval=1:~success#CCancelled;;  
select top 1 * from diary where description like 'Shift Candidate Rejected%'  
order by whenentered desc;  
expect  
staffid=TEST,personid=test,vacancyID=TEST,EmploymentID=TEST,notes=TestNote;  
*/  
}
```

From:

<https://iqxusers.co.uk/iqxhelp/> - iqx

Permanent link:

https://iqxusers.co.uk/iqxhelp/doku.php?id=database:procedures:pears_netclientvacancieslinestatusupdate

Last update: **2026/08/07 19:24**

