



pears.NetProvTimesheetShiftSet



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

Original SQL

```
CREATE PROCEDURE "pears"."NetProvTimesheetShiftSet"( IN "pWebUserID"
CHAR(20),IN "ptempshiftid" CHAR(250),IN "pshiftdate" CHAR(250) DEFAULT
'NOT_SET',IN "ptimefrom" CHAR(250) DEFAULT 'NOT_SET',IN "ptimeto" CHAR(250)
DEFAULT 'NOT_SET',IN "pbreakminutes" CHAR(250) DEFAULT 'NOT_SET',IN
"preferencecode" CHAR(20) DEFAULT NULL,IN "ptempshifttypeid" CHAR(2) DEFAULT
NULL )
RESULT( "pResult" CHAR(250) )
// IQXWeb
BEGIN
    DECLARE "i" SMALLINT;
    DECLARE "isnewshift" SMALLINT;
    DECLARE "newshiftid" CHAR(20);
    DECLARE "userClass" CHAR(20);
    DECLARE "divid" CHAR(20);
    DECLARE "checkstart" DATE;
    DECLARE "checkend" DATE;
    DECLARE LOCAL TEMPORARY TABLE "IDs"(
        "ID" CHAR(20) NULL,
    ) NOT transactional;
    IF "trim"("pbreakminutes") = '' THEN
        SET "pbreakminutes" = '0'
    END IF;
    SET "preferencecode" = "ucase"("preferencecode");
    SET "ptempshifttypeid" = "nullif"("trim"("ptempshifttypeid"), '');
    IF "ptempshiftid" LIKE 'Copy%' THEN
        SET "isnewshift" = 1
    ELSE
        SET "isnewshift" = 0
    END IF;
    SET "i" = "charindex"('_', "ptempshiftid");
    IF "i" > 0 THEN
        SET "ptempshiftid" = "right"("ptempshiftid", "length"("ptempshiftid") -
        "i")
    END IF;
    SET "userClass" = (SELECT FIRST "iqxnetuserclassid" FROM "iqxnetuser"
```



```
WHERE "iqxnetuserid" = "pWebUserID");
  IF "userClass" = 'OWNER' THEN
    INSERT INTO "IDs"( "ID" )
      SELECT DISTINCT "s"."divisionid" FROM "iqxnetuser" AS "i" KEY JOIN
"staff" AS "s" WHERE "i"."iqxnetuserid" = "pwebuserid";
    IF NOT "ptempshiftid" = any(SELECT "s"."tempshiftid" FROM "tempshift" AS
"s" KEY JOIN "person" AS "p" JOIN "IDs" ON "p"."divisionid" = "IDs"."ID")
  THEN
    SELECT '99:~Permission denied';
    RETURN
  END IF
  ELSE IF NOT "pTempShiftID" = any(
    SELECT "t"."TempShiftID" FROM "tempshift" AS "t" KEY JOIN "person" KEY
JOIN "iqxnetuserlink" WHERE "iqxnetuserid" = "pwebuserid" UNION
    SELECT "t"."TempShiftID" FROM "tempshift" AS "t" KEY JOIN "person" KEY
JOIN "pay_employee" KEY JOIN "company" AS "agcomp" KEY JOIN "employment" AS
"agemp" KEY JOIN "iqxnetuserlink"
      WHERE "iqxnetuserlink"."iqxnetuserid" = "pwebuserid" UNION
    SELECT "t"."TempShiftID" FROM "tempshift" AS "t" KEY JOIN "vacancy"
KEY JOIN "employment" AS "vacemp" KEY JOIN "company" KEY JOIN "employment"
KEY JOIN "iqxnetuserlink"
      WHERE "iqxnetuserlink"."iqxnetuserid" = "pwebuserid") THEN
    SELECT '99:~Permission denied';
    RETURN
  END IF END IF;
  IF "isnewshift" = 1 THEN
    SET "newshiftid" = "uniquekey"("ptempshiftid");
    IF "trim"("preferencecode") = '' THEN
      SET "preferencecode" = NULL
    END IF;
    IF "isnull"("IQXNetSwitchValue"("pwebuserid",'SHIFTDATELIMIT'),0) = 1
  THEN
    SELECT
    "e"."startdate","dateformat"("isnull"("e"."leavedate","dateadd"("year",1,"to
day"()),'yyyy-mm-dd') INTO "checkstart","checkend" FROM "tempshift" KEY
JOIN "placement" AS "p" KEY JOIN "employment" AS "e" WHERE "tempshiftid" =
"ptempshiftid";
    IF "iqxnetstringtodate"("pShiftDate") NOT BETWEEN "checkstart" AND
"checkend" THEN
      SELECT '103:~Date must be between
'+"dateformat"("checkstart",'dd/mm/yyyy')+' and
'+"dateformat"("checkend",'dd/mm/yyyy');
      RETURN
    END IF END IF;
    INSERT INTO "tempshiftplan"(
    "TempShiftPlanID","VacancyID","ShiftDate","TimeFrom","TimeTo","BreakMinutes"
    ,"Description","EssentialSkill","EssentialSkillGradeID","ReferenceCode",
    "EssentialSkillChoiceList","TempShiftTypeID","AnalysisCode","RecoveryHours"
```



```
)  
    SELECT  
    "newshiftid","VacancyID","isnull"("iqxnetstringtodate"("pshiftdate"),"shiftdate"),  
    "isnull"("iqxnetstringtotime"("ptimefrom"),"timefrom"),  
    "isnull"("iqxnetstringtotime"("ptimeto"),"timeto"),"isnull"("iqxnetstringtointeger"("pbreakminutes"),  
    "breakminutes"),"Description","EssentialSkill","EssentialSkillGradeID","isnull"("preferencecode",  
    'Self-booked'),  
    "EssentialSkillChoiceList","isnull"("ptempshifttypeid","TempShiftTypeID"),"AnalysisCode",  
    "RecoveryHours" FROM "tempshiftplan"  
    WHERE "tempshiftplanid" = (SELECT "tempshiftplanid" FROM "tempshift"  
    WHERE "tempshiftid" = "ptempshiftid");  
    INSERT INTO "tempshift"(  
    "TempShiftID","VacancyID","PersonID","PlacementID","ShiftDate","TimeFrom","TimeTo",  
    "BreakMinutes","State","TempShiftPlanID","EssentialSkillGradeID","ReferenceCode",  
    "ClientConfirmed","TempConfirmed","TempShiftTypeID","AnalysisCode","RecoveryHours" )  
    SELECT  
    "NewShiftID","VacancyID","PersonID","PlacementID","isnull"("iqxnetstringtodate"("pshiftdate"),  
    "shiftdate"),"isnull"("iqxnetstringtotime"("ptimefrom"),"timefrom"),  
    "isnull"("iqxnetstringtotime"("ptimeto"),"timeto"),"isnull"("iqxnetstringtointeger"("pbreakminutes"),  
    "breakminutes"),'B',"NewShiftID","EssentialSkillGradeID","isnull"("preferencecode",  
    'Self-booked'),  
    1,1,"isnull"("ptempshifttypeid","TempShiftTypeID"),"AnalysisCode","RecoveryHours" FROM "tempshift"  
    WHERE "tempshiftid" = "ptempshiftid"  
    ELSE  
    UPDATE "tempshift" SET "shiftdate" =  
    "isnull"("iqxnetstringtodate"("pshiftdate"),"shiftdate"),"timefrom" =  
    "isnull"("iqxnetstringtotime"("ptimefrom"),"timefrom"),  
    "timeto" =  
    "isnull"("iqxnetstringtotime"("ptimeto"),"timeto"),"breakminutes" =  
    "isnull"("iqxnetstringtointeger"("pbreakminutes"),"breakminutes"),  
    "ReferenceCode" =  
    "isnull"("preferencecode","ReferenceCode"),"TempShiftTypeID" =  
    "isnull"("ptempshifttypeid","TempShiftTypeID")  
    WHERE "tempshiftid" = "ptempshiftid"  
    END IF;  
    SELECT '0:~Success'  
END /* DOC  
2016-12-07 PC doc and test  
2018-02-26 PC IW-381 check shift date between employment start and end dates  
2018-07-26 PC include improved error handler  
2018-09-05 ET changed back to old way of returning errors, updated error  
number so that system errors are < 100 and user errors >= 100  
2018-10-11 PC IW-705 reverse IW-381
```



```
Tests:
invalid pWebUserID
valid pWebUserID, invalid pTempShiftID as owner
valid pWebUserID, invalid pTempShiftID as client
valid pWebUserID, pTempShiftID, new shift as client
valid pWebUserID, pTempShiftID, copy shift as client

*/
/* TEST
call NetTestSetup('');
update staff set divisionid='TEST' where staffid='TEST';
update person set divisionid='TEST' where personid='TEST';
update tempshift set personid='TEST' where tempshiftid='TEST';
select * from
NetProvTimesheetShiftSet(null,null,null,null,null,null,null,null);
expect 99*;
select * from
NetProvTimesheetShiftSet('test.owner',null,null,null,null,null,null,null);
expect 0*;
select * from
NetProvTimesheetShiftSet('test.owner','Shift_TEST',null,null,null,null,null,
null);
expect 0*;

*/
GO

COMMENT TO PRESERVE FORMAT ON PROCEDURE "pears"."NetProvTimesheetShiftSet"
IS
{CREATE PROCEDURE pears."NetProvTimesheetShiftSet"(IN pWebUserID CHAR(20),IN
ptempshiftid CHAR(250),IN pshiftdate CHAR(250) DEFAULT 'NOT_SET',IN
ptimefrom CHAR(250) DEFAULT 'NOT_SET',IN ptimeto CHAR(250) DEFAULT
'NOT_SET',IN pbreakminutes CHAR(250) DEFAULT 'NOT_SET', IN preferencecode
CHAR(20) DEFAULT NULL, IN ptempshifttypeid CHAR(2) DEFAULT NULL)
RESULT(pResult CHAR(250))
// IQXWeb
BEGIN
  DECLARE i SMALLINT;
  DECLARE isnewshift SMALLINT;
  DECLARE newshiftid CHAR(20);
  DECLARE userClass CHAR(20);
  DECLARE divid CHAR(20);
  DECLARE checkstart DATE;
  DECLARE checkend DATE;
  DECLARE LOCAL TEMPORARY TABLE IDs(
    ID CHAR(20) NULL,
  ) NOT transactional;
  IF TRIM(pbreakminutes) = '' THEN
```



```
SET pbreakminutes='0'
END IF;
SET preferencecode=ucase(preferencecode);
SET ptempshifttypeid=NULLIF(TRIM(ptempshifttypeid),'');
IF ptempshiftid LIKE 'Copy%' THEN
    SET isnewshift=1
ELSE
    SET isnewshift=0
END IF;
SET i=charindex('_',ptempshiftid);
IF i > 0 THEN
    SET ptempshiftid="right"(ptempshiftid,LENGTH(ptempshiftid)-i)
END IF;
SET userClass=(SELECT FIRST iqxnetuserclassid FROM iqxnetuser WHERE
iqxnetuserid = pWebUserID);
IF userClass = 'OWNER' THEN
    INSERT INTO IDs( ID)
        SELECT DISTINCT s.divisionid FROM iqxnetuser AS i KEY JOIN staff AS s
WHERE i.iqxnetuserid = pwebuserid;
    IF NOT ptempshiftid = any(SELECT s.tempshiftid FROM tempshift AS s KEY
JOIN person AS p JOIN IDs ON p.divisionid = IDs.ID) THEN
        SELECT '99:~Permission denied';
        RETURN
    END IF
ELSE
    IF NOT pTempShiftID = any(
        SELECT t.TempShiftID FROM tempshift AS t KEY JOIN person KEY JOIN
iqxnetuserlink WHERE iqxnetuserid = pwebuserid UNION
        SELECT t.TempShiftID FROM tempshift AS t KEY JOIN person KEY JOIN
pay_employee KEY JOIN company AS agcomp KEY JOIN employment AS agemp KEY
JOIN iqxnetuserlink
        WHERE iqxnetuserlink.iqxnetuserid = pwebuserid UNION
        SELECT t.TempShiftID FROM tempshift AS t KEY JOIN vacancy KEY JOIN
employment AS vacemp KEY JOIN company KEY JOIN employment KEY JOIN
iqxnetuserlink
        WHERE iqxnetuserlink.iqxnetuserid = pwebuserid) THEN
        SELECT '99:~Permission denied';
        RETURN
    END IF
END IF;
IF isnewshift = 1 THEN
    SET newshiftid=uniquekey(ptempshiftid);
    IF TRIM(preferencecode)='' THEN
        SET preferencecode=NULL
    END IF;
    IF isnull(IQXNetSwitchValue(pwebuserid,'SHIFTDATELIMIT'),0) = 1 THEN
        SELECT e.startdate,
dateformat(isnull(e.leavedate,dateadd(YEAR,1,today(*))), 'yyyy-mm-dd') INTO
```



```
checkstart,checkend FROM tempshift KEY JOIN placement p KEY JOIN employment
e WHERE tempshiftid=ptempshiftid;
    IF iqxnetstringtodate(pShiftDate) NOT BETWEEN checkstart AND
checkend THEN
        SELECT '103:~Date must be between
'+dateformat(checkstart,'dd/mm/yyyy')+' and
'+dateformat(checkend,'dd/mm/yyyy');
        RETURN
    END IF;
END IF;
INSERT INTO tempshiftplan(
TempShiftPlanID,VacancyID,ShiftDate,TimeFrom,TimeTo,BreakMinutes,Description
,EssentialSkill,EssentialSkillGradeID,ReferenceCode,
    EssentialSkillChoiceList,TempShiftTypeID,AnalysisCode,RecoveryHours)
SELECT
newshiftid,VacancyID,isnull(iqxnetstringtodate(pshiftdate),shiftdate),isnull
(iqxnetstringtotime(ptimefrom),timefrom),
isnull(iqxnetstringtotime(ptimeto),timeto),isnull(iqxnetstringtointeger(pbre
akminutes),breakminutes),Description,EssentialSkill,EssentialSkillGradeID,is
null(preferencencode,'Self-booked'),
EssentialSkillChoiceList,isnull(ptempshifttypeid,TempShiftTypeID),AnalysisCo
de,RecoveryHours FROM tempshiftplan
    WHERE tempshiftplanid = (SELECT tempshiftplanid FROM tempshift WHERE
tempshiftid = ptempshiftid);
INSERT INTO tempshift(
TempShiftID,VacancyID,PersonID,PlacementID,ShiftDate,TimeFrom,TimeTo,BreakMi
nutes,State,TempShiftPlanID,EssentialSkillGradeID,ReferenceCode,
ClientConfirmed,TempConfirmed,TempShiftTypeID,AnalysisCode,RecoveryHours)
SELECT
NewShiftID,VacancyID,PersonID,PlacementID,isnull(iqxnetstringtodate(pshiftda
te),shiftdate),isnull(iqxnetstringtotime(ptimefrom),timefrom),
isnull(iqxnetstringtotime(ptimeto),timeto),isnull(iqxnetstringtointeger(pbre
akminutes),breakminutes),'B',NewShiftID,EssentialSkillGradeID,isnull(prefere
ncencode,'Self-booked'),
1,1,isnull(ptempshifttypeid,TempShiftTypeID),AnalysisCode,RecoveryHours FROM
tempshift
    WHERE tempshiftid = ptempshiftid
ELSE
    UPDATE tempshift SET shiftdate =
isnull(iqxnetstringtodate(pshiftdate),shiftdate),timefrom =
isnull(iqxnetstringtotime(ptimefrom),timefrom),
        timeto = isnull(iqxnetstringtotime(ptimeto),timeto),breakminutes =
isnull(iqxnetstringtointeger(pbreakminutes),breakminutes),
        ReferenceCode = isnull(preferencencode,ReferenceCode), TempShiftTypeID
= isnull(ptempshifttypeid,TempShiftTypeID)
        WHERE tempshiftid = ptempshiftid
END IF;
SELECT '0:~Success'
```

**END**

```
/* DOC
2016-12-07 PC doc and test
2018-02-26 PC IW-381 check shift date between employment start and end dates
2018-07-26 PC include improved error handler
2018-09-05 ET changed back to old way of returning errors, updated error
number so that system errors are < 100 and user errors >= 100
2018-10-11 PC IW-705 reverse IW-381
```

Tests:

```
    invalid pWebUserID
    valid pWebUserID, invalid pTempShiftID as owner
    valid pWebUserID, invalid pTempShiftID as client
    valid pWebUserID, pTempShiftID, new shift as client
    valid pWebUserID, pTempShiftID, copy shift as client

*/

/* TEST
call NetTestSetup('');
update staff set divisionid='TEST' where staffid='TEST';
update person set divisionid='TEST' where personid='TEST';
update tempshift set personid='TEST' where tempshiftid='TEST';
select * from
NetProvTimesheetShiftSet(null,null,null,null,null,null,null,null);
expect 99*;
select * from
NetProvTimesheetShiftSet('test.owner',null,null,null,null,null,null,null);
expect 0*;
select * from
NetProvTimesheetShiftSet('test.owner','Shift_TEST',null,null,null,null,null,
null);
expect 0*;

*/
}
```

From:

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

Permanent link:

https://iqxusers.co.uk/iqxhelp/doku.php?id=database:procedures:pears_netprovtimesheetshiftsetLast update: **2026/08/07 19:24**