



pears.NetOwnerDiarySet



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

Original SQL

```
CREATE PROCEDURE "pears"."NetOwnerDiarySet"( IN "pWebUserID" CHAR(20),IN
"pDiaryID" CHAR(20) DEFAULT NULL,IN "pAction" CHAR(1),IN "pEventType"
CHAR(1) DEFAULT NULL,IN "pStaffID" CHAR(20) DEFAULT NULL,IN "pPersonID"
CHAR(20) DEFAULT NULL,IN "pEmploymentID" CHAR(20) DEFAULT NULL,IN
"pStartDate" CHAR(20) DEFAULT NULL,IN "pEndDate" CHAR(20) DEFAULT NULL,IN
"pStartTime" CHAR(20) DEFAULT NULL,IN "pEndTime" CHAR(20) DEFAULT NULL,IN
"pDescription" CHAR(100) DEFAULT NULL,IN "pNotes" long VARCHAR DEFAULT
NULL,IN "pPriority" CHAR(1) DEFAULT NULL,IN "pMinutesBefore" INTEGER DEFAULT
NULL )
RESULT( "pResult" CHAR(250) )
BEGIN
  // IQXWeb
  DECLARE "dEndTime" "datetime";
  DECLARE "dStartTime" "datetime";
  IF "pAction" NOT IN( 'N','U','D' ) THEN
    SELECT '3::~Invalid pAction';
    RETURN
  END IF;
  IF "pEventType" NOT IN( 'A','R' ) AND "pAction" = 'U' THEN
    SELECT '6::~Invalid Event Type';
    RETURN
  END IF;
  IF "isnull"("pDiaryID",'') = '' AND "pAction" IN( 'U','D' ) THEN -- must
have a DiaryID for update or delete
    SELECT '2::~pDiaryID required';
    RETURN
  END IF;
  IF "pAction" <> 'D' THEN
    IF "isnull"("pStartDate",'') <> '' THEN
      SET "dStartTime" =
"iqxnetstringtodate"("pstartdate")+"iqxnetstringtotime"("isnull"("pstarttime",
'00:00'));
      IF "dStartTime" IS NULL THEN
        SELECT '107::~Invalid start date or time';
        RETURN
      END IF;
    END IF;
  END IF;
END
```



```
END IF END IF;
IF "isnull"("pEndDate",'') <> '' THEN
  SET "dEndDateTime" =
  "iqxnetstringtodate"("penddate")+"iqxnetstringtotime"("isnull"("pendtime",'0
0:00'));
  IF "dEndDateTime" IS NULL THEN
    SELECT '107:~Invalid end date or time';
    RETURN
  END IF END IF;
IF "dEndDateTime" < "dStartDateTime" THEN
  SELECT '108:~Finish time appears to be before Start time';
  RETURN
END IF;
IF "pAction" = 'N' AND "dEndDateTime" IS NULL THEN
  SET "dEndDateTime" = "dStartDateTime"
END IF END IF;
CASE "pAction"
WHEN 'N' THEN
  IF "dStartDateTime" IS NULL THEN
    SELECT '104:~pStartDate required';
    RETURN
  END IF;
  IF "trim"("isnull"("pDescription",'')) = '' THEN
    SELECT '105:~pDescription required';
    RETURN
  END IF;
  IF "pMinutesBefore" > 30240 THEN
    SELECT '106:~Minutes Before cannot exceed 30240 minutes (or 21 days)';
    RETURN
  END IF;
  IF "isnull"("pStaffID",'') = '' THEN
    SET "pStaffID" = (SELECT "staffid" FROM "iqxnetuser" WHERE
"iqxnetuserid" = "pWebUserID")
  END IF;
  INSERT INTO "diary"(
"diaryid","staffid","personid","employmentid","diaryfrom","diaryto","descrip
tion","notes","priority","PopupMinutes" ) VALUES
(
"isnull"("pDiaryID","uniquekey"('x')), "pStaffID", "pPersonID", "pEmploymentID"
,"dStartDateTime", "dEndDateTime", "pDescription", "pNotes", "pPriority", "pMinut
esBefore" ) ;
  SELECT '0:~Success';
  RETURN
WHEN 'U' THEN -- Update record
  IF "pEventType" = 'R' THEN
    SET "dEndDateTime" = "dStartDateTime"
  END IF;
  UPDATE "diary" SET "staffid" = "isnull"("pStaffID","staffid"),"personid"
```



```
= "isnull"("pPersonID","personid"),"employmentid" =
"isnull"("pEmploymentID","employmentid"),"diaryfrom" =
"isnull"("dStartDateTime","diaryfrom"),"diaryto" =
"isnull"("dEndDateTime","diaryto"),"description" =
"isnull"("pDescription","description"),"notes" =
"isnull"("pNotes","notes"),"priority" =
"isnull"("pPriority","priority"),"durationtype" = NULL,"PopupMinutes" =
"pMinutesBefore","PopupTime" = "dateadd"("minute",0-
"pMinutesBefore","DiaryFrom") WHERE "DiaryID" = "pDiaryID";
-- Note setting durationtype=null to prevent the trigger mucking up the
diaryto based on previous stored duration
SELECT '0:~Success';
RETURN
WHEN 'D' THEN -- delete record
DELETE FROM "diary" WHERE "diaryid" = "pDiaryID";
SELECT '0:~Success';
RETURN
END CASE;
SELECT '1:~Error' -- if here then no action has been specified
END /* TEST
call NetTestSetup('');
delete from diary where staffid='test';
select * from NetOwnerDiarySet(pWebUserID='test.owner', pAction='N',
pStartDate='25/12/2001', pStartTime='09:00', pDescription='TEST991');
expect pResult=0:*;
select * from diary where staffid='TEST' and Description='TEST991';
expect diaryfrom=25/12/2001 09:00*, durationtype=R;
select * from NetOwnerDiarySet(pWebUserID='test.owner', pDiaryID='TEST992',
pAction='N', pStartDate='24/12/2001', pEndDate='25/12/2001',
pDescription='TEST992');
expect pResult=0:*;
select * from diary where DiaryID='TEST992';
expect diaryfrom=24/12/2001*, durationtype=D;
call NetOwnerDiarySet(pWebUserID='test.owner', pDiaryID='TEST992',
pAction='U', pNotes='TestNotes', pStartDate='23/12/2001',
pStartTime='09:15');
select * from diary where DiaryID='TEST992';
expect notes=TestNotes,diaryfrom=23/12/2001 09:15*,diaryto=25/12/2001;
call NetOwnerDiarySet(pWebUserID='test.owner', pDiaryID='TEST992',
pAction='U', pNotes='TestNotes', pStartDate='23/12/2001',
pStartTime='09:00',pMinutesBefore='30');
select notes,replace(diaryfrom,' ','!') as
DF,diaryto,popupminutes,replace(popuptime,' ','!') as PT from diary where
DiaryID='TEST992';
expect
notes=TestNotes,DF=2001-12-23!09:00:00.000000,diaryto=25/12/2001,popupminute
s=30,PT=2001-12-23!08:30:00.000000;
call NetOwnerDiarySet(pWebUserID='test.owner', pDiaryID='TEST992',
```



```
pAction='D');
select notes,replace(diaryfrom,' ','!') as
DF,diaryto,popupminutes,replace(popuptime,' ','!') as PT from diary where
DiaryID='TEST992';
expect EOF;
select * from NetOwnerDiarySet(pWebUserID='test.owner', pAction='Z',
pStartDate='25/12/2001', pStartTime='09:00', pDescription='TEST991');
expect pResult=3:*;
select * from NetOwnerDiarySet(pWebUserID='test.owner', pAction='N',
pStartDate=null, pStartTime='09:00', pDescription='TEST991');
expect pResult=4:*;
select * from NetOwnerDiarySet(pWebUserID='test.owner', pAction='N',
pStartDate='25/12/2001', pStartTime='09:00');
expect pResult=5:*;
select * from NetOwnerDiarySet(pWebUserID='test.owner', pAction='N',
pStartDate='25/12/2001',
pStartTime='09:00',pEndDate='24/12/2001',pEndTime='08:00');
expect pResult=8:*;
select * from NetOwnerDiarySet(pWebUserID='test.owner', pAction='D');
expect pResult=2:*;
delete from diary where staffid='test';
*/
/* DOC
2016-07-26 PC Provide new/update/delete facility for NetOwnerDiary
2016-09-07 GJ When editing a REMINDER set the end datetime to the same as
the start date time
2016-09-27 PC modified to accept dates with slash
2016-09-30 PC MS Modify logic to rationalise validation
2016-11-10 PC Modify date validation to cover validation removed from web
page
2017-10-30 GJ There is now a 21 day limit on the MinutesBefore popup to
match what IQX.exe does
2018-05-14 ET Removed the bit setting 'who entered' to staff ID as this was
causing it to be set to the wrong staff ID, not the staff ID of the person
who actually entered it.
```

Incoming:

pAction - N for New, U for Update, D for Delete

pDiaryID - PK of line to update or delete - required for update and delete but not new

pStaffID - staff member to make appointment with - defaults to logged in user

pPersonID - candidate to make appointment with - optional

pEmploymentID - contact to make appointment with - optional

pStartDate and pStartTime - start date and time for appointment or reminder - date required time optional

pEndDate and pEndTime - end date and time for appointment - if not supplied it is a reminder



pDescription - title of diary entry - required
pNotes - notes for diary entry - optional
pPriority - priority level for diary entry - optional
Outgoing:
pResult - success or error message

Tests:

create a reminder
create an appointment
change the appointment notes
delete the appointment
check for error reports on invalid action, corrupt date, missing description
and missing pDiaryID
*/
GO

```
COMMENT TO PRESERVE FORMAT ON PROCEDURE "pears"."NetOwnerDiarySet" IS
{CREATE PROCEDURE pears."NetOwnerDiarySet"(IN pWebUserID CHAR(20),IN
pDiaryID CHAR(20) DEFAULT NULL,IN pAction CHAR(1),IN pEventType CHAR(1)
DEFAULT NULL,IN pStaffID CHAR(20) DEFAULT NULL,IN pPersonID CHAR(20) DEFAULT
NULL,IN pEmploymentID CHAR(20) DEFAULT NULL,IN pStartDate CHAR(20) DEFAULT
NULL,IN pEndDate CHAR(20) DEFAULT NULL,pStartTime CHAR(20) DEFAULT
NULL,pEndTime CHAR(20) DEFAULT NULL,IN pDescription CHAR(100) DEFAULT
NULL,IN pNotes long VARCHAR DEFAULT NULL,IN pPriority CHAR(1) DEFAULT
NULL,pMinutesBefore INTEGER DEFAULT NULL)
RESULT( "pResult" CHAR(250) )
BEGIN
// IQXWeb
DECLARE dEndTime datetime;
DECLARE dStartTime datetime;

IF NOT pAction IN ('N','U','D') THEN
    SELECT '3:~Invalid pAction';
    RETURN;
END IF;
IF NOT pEventType IN ('A','R') AND pAction='U' THEN
    SELECT '6:~Invalid Event Type';
    RETURN;
END IF;
IF isnull(pDiaryID,'')='' AND pAction IN ('U','D') THEN -- must have a
DiaryID for update or delete
    SELECT '2:~pDiaryID required';
    RETURN;
END IF;

IF pAction !='D' THEN
    IF isnull(pStartDate,'') <>' ' THEN
        SET
```



```
dStartDateTime=iqxnetstringtodate(pstartdate)+iqxnetstringtotime(isnull(psta
rttime,'00:00'));
    IF dStartDateTime IS NULL THEN
        SELECT '107:~Invalid start date or time';
        RETURN;
    END IF;
END IF;

IF isnull(pEndDate,'') <> '' THEN
    SET
dEndDateTime=iqxnetstringtodate(penddate)+iqxnetstringtotime(isnull(pendtime
,'00:00'));
    IF dEndDateTime IS NULL THEN
        SELECT '107:~Invalid end date or time';
        RETURN;
    END IF;
END IF;

IF dEndDateTime < dStartDateTime THEN
    SELECT '108:~Finish time appears to be before Start time';
    RETURN;
END IF;

IF pAction='N' AND dEndDateTime IS NULL THEN
    SET dEndDateTime=dStartDateTime
END IF;
END IF;

CASE pAction
WHEN 'N' THEN
    IF dStartDateTime IS NULL THEN
        SELECT '104:~pStartDate required';
        RETURN;
    END IF;
    IF TRIM(isnull(pDescription,''))='' THEN
        SELECT '105:~pDescription required';
        RETURN;
    END IF;
    IF pMinutesBefore > 30240 THEN
        SELECT '106:~Minutes Before cannot exceed 30240 minutes (or 21
days)';
        RETURN;
    END IF;
    IF isnull(pStaffID,'')='' THEN
        SET pStaffID=(SELECT staffid FROM iqxnetuser WHERE
iqxnetuserid=pWebUserID);
    END IF;
    INSERT INTO diary
```



```
(diaryid,staffid,personid,employmentid,diaryfrom,diaryto,description,notes,p
riority,PopupMinutes)
VALUES
(isnull(pDiaryID,uniquekey('x')),pStaffID,pPersonID,pEmploymentID,dStartDate
Time,dEndDateTime,pDescription,pNotes,pPriority,pMinutesBefore);
SELECT '0:~Success';
RETURN;
WHEN 'U' THEN -- Update record
IF pEventType='R' THEN
SET dEndDateTime=dStartDateTime
END IF;
UPDATE diary SET staffid=isnull(pStaffID,staffid),
personid=isnull(pPersonID,personid),
employmentid=isnull(pEmploymentID,employmentid),
diaryfrom=isnull(dStartDateTime,diaryfrom),
diaryto=isnull(dEndDateTime,diaryto),
description=isnull(pDescription,description),
notes=isnull(pNotes,notes),priority=isnull(pPriority,priority),
durationtype=NULL,PopupMinutes=pMinutesBefore,PopupTime=dateadd(MINUTE,0-
pMinutesBefore,DiaryFrom) WHERE DiaryID=pDiaryID;
-- Note setting durationtype=null to prevent the trigger mucking up the
diaryto based on previous stored duration
SELECT '0:~Success';
RETURN
WHEN 'D' THEN -- delete record
DELETE FROM diary WHERE diaryid=pDiaryID;
SELECT '0:~Success';
RETURN
END CASE;
SELECT '1:~Error'; -- if here then no action has been specified
END

/* TEST
call NetTestSetup('');
delete from diary where staffid='test';
select * from NetOwnerDiarySet(pWebUserID='test.owner', pAction='N',
pStartDate='25/12/2001', pStartTime='09:00', pDescription='TEST991');
expect pResult=0:*;
select * from diary where staffid='TEST' and Description='TEST991';
expect diaryfrom=25/12/2001 09:00*, durationtype=R;
select * from NetOwnerDiarySet(pWebUserID='test.owner', pDiaryID='TEST992',
pAction='N', pStartDate='24/12/2001', pEndDate='25/12/2001',
pDescription='TEST992');
expect pResult=0:*;
select * from diary where DiaryID='TEST992';
expect diaryfrom=24/12/2001*, durationtype=D;
call NetOwnerDiarySet(pWebUserID='test.owner', pDiaryID='TEST992',
pAction='U', pNotes='TestNotes', pStartDate='23/12/2001',
```



```
pStartTime='09:15');
select * from diary where DiaryID='TEST992';
expect notes=TestNotes,diaryfrom=23/12/2001 09:15*,diaryto=25/12/2001;
call NetOwnerDiarySet(pWebUserID='test.owner', pDiaryID='TEST992',
pAction='U', pNotes='TestNotes', pStartDate='23/12/2001',
pStartTime='09:00',pMinutesBefore='30');
select notes,replace(diaryfrom,' ','!') as
DF,diaryto,popupminutes,replace(popuptime,' ','!') as PT from diary where
DiaryID='TEST992';
expect
notes=TestNotes,DF=2001-12-23!09:00:00.000000,diaryto=25/12/2001,popupminute
s=30,PT=2001-12-23!08:30:00.000000;
call NetOwnerDiarySet(pWebUserID='test.owner', pDiaryID='TEST992',
pAction='D');
select notes,replace(diaryfrom,' ','!') as
DF,diaryto,popupminutes,replace(popuptime,' ','!') as PT from diary where
DiaryID='TEST992';
expect EOF;
select * from NetOwnerDiarySet(pWebUserID='test.owner', pAction='Z',
pStartDate='25/12/2001', pStartTime='09:00', pDescription='TEST991');
expect pResult=3:*;
select * from NetOwnerDiarySet(pWebUserID='test.owner', pAction='N',
pStartDate=null, pStartTime='09:00', pDescription='TEST991');
expect pResult=4:*;
select * from NetOwnerDiarySet(pWebUserID='test.owner', pAction='N',
pStartDate='25/12/2001', pStartTime='09:00');
expect pResult=5:*;
select * from NetOwnerDiarySet(pWebUserID='test.owner', pAction='N',
pStartDate='25/12/2001',
pStartTime='09:00',pEndDate='24/12/2001',pEndTime='08:00');
expect pResult=8:*;
select * from NetOwnerDiarySet(pWebUserID='test.owner', pAction='D');
expect pResult=2:*;
delete from diary where staffid='test';
*/

/* DOC
2016-07-26 PC Provide new/update/delete facility for NetOwnerDiary
2016-09-07 GJ When editing a REMINDER set the end datetime to the same as
the start date time
2016-09-27 PC modified to accept dates with slash
2016-09-30 PC MS Modify logic to rationalise validation
2016-11-10 PC Modify date validation to cover validation removed from web
page
2017-10-30 GJ There is now a 21 day limit on the MinutesBefore popup to
match what IQX.exe does
2018-05-14 ET Removed the bit setting 'who entered' to staff ID as this was
causing it to be set to the wrong staff ID, not the staff ID of the person
```




who actually entered it.

Incoming:

pAction - N for New, U for Update, D for Delete

pDiaryID - PK of line to update or delete - required for update and delete but not new

pStaffID - staff member to make appointment with - defaults to logged in user

pPersonID - candidate to make appointment with - optional

pEmploymentID - contact to make appointment with - optional

pStartDate and *pStartTime* - start date and time for appointment or reminder - date required time optional

pEndDate and *pEndTime* - end date and time for appointment - if not supplied it is a reminder

pDescription - title of diary entry - required

pNotes - notes for diary entry - optional

pPriority - priority level for diary entry - optional

Outgoing:

pResult - success or error message

Tests:

create a reminder

create an appointment

change the appointment notes

delete the appointment

check for error reports on invalid action, corrupt date, missing description and missing pDiaryID

**/*

}

From:

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

Permanent link:

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

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

