



pears.NetOwnerContactEventsSet



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

Original SQL

```
CREATE PROCEDURE "pears"."NetOwnerContactEventsSet"( IN "pWebUserID"
CHAR(20),IN "pContactEventID" CHAR(20) DEFAULT NULL,IN "pAction" CHAR(1),IN
"pClassCode" CHAR(5) DEFAULT NULL,IN "pPersonID" CHAR(20) DEFAULT NULL,IN
"pEmploymentID" CHAR(20) DEFAULT NULL,IN "pVacancyID" CHAR(20) DEFAULT
NULL,IN "pEventDate" CHAR(20) DEFAULT NULL,IN "pDescription" CHAR(100)
DEFAULT NULL,IN "pNotes" long VARCHAR DEFAULT NULL,IN "pCallbackDate"
CHAR(20) DEFAULT NULL,IN "pOutcome" CHAR(20) DEFAULT NULL,IN "pPriority"
CHAR(1) DEFAULT NULL )
RESULT( "pResult" CHAR(250) )
BEGIN
  DECLARE "dEventDate" CHAR(20);
  DECLARE "dContactDate" CHAR(10);
  DECLARE "dContactTime" TIME;
  DECLARE "pCallbackText" CHAR(20);
  DECLARE "dCallbackDate" CHAR(10);
  DECLARE "dCallbackTime" TIME;
  DECLARE "pStaffID" CHAR(20);
  DECLARE "pDivisionID" CHAR(20);
  IF "isnull"("pEventDate",'') <> '' THEN
    SET "dEventDate" = "IQXNetStringToDate"("left"("pEventDate",10));
    SET "pEventDate" = "stuff"("pEventDate",1,11,'');
    SET "dContactTime" = "IQXNetStringToTime"("pEventDate")
  ELSE
    SET "dEventDate" = "IQXNetStringToDate"("left"("now"(),10));
    SET "dContactTime" = NULL
  END IF;
  SET "dContactDate" = "dEventDate";
  IF "dEventDate" IS NULL AND "pAction" = 'N' THEN
    SELECT '4:~Invalid Event date';
    RETURN
  END IF;
  IF "isnull"("pCallbackDate",'') <> '' THEN
    SET "dCallbackDate" = "IQXNetStringToDate"("left"("pCallbackDate",10));
    SET "pCallbackDate" = "stuff"("pCallbackDate",1,11,'');
    SET "dCallbackTime" = "IQXNetStringToTime"("pCallbackDate")
```



```
ELSE
    SET "dCallbackDate" = NULL;
    SET "dCallbackTime" = NULL
END IF;
SET "pStaffID" = (SELECT "staffid" FROM "iqxnetuser" WHERE "iqxnetuserid"
= "pWebUserID");
SET "pDivisionID" = (SELECT "divisionid" FROM "staff" WHERE "staffid" =
"pStaffID");
SET "pCallbackText" = (SELECT top 1 "description" FROM "contactoutcome"
WHERE "contactoutcomeid" = "pOutcome");
IF "pAction" NOT IN( 'N','U','D' ) THEN
    SELECT '3:~Invalid Action';
    RETURN
END IF;
IF "isnull"("pContactEventID",'') = '' AND "pAction" IN( 'U','D' ) THEN --
must have a ContactEventID for update or delete
    SELECT '2:~pContactEventID required';
    RETURN
END IF;
IF "pAction" = 'N' THEN
    IF "pClassCode" IS NULL THEN
        SELECT '8:~Invalid Event code';
        RETURN
    END IF;
    IF NOT "pClassCode" = any(SELECT "classcode" FROM "contactclass" WHERE
"classcode" = "pClassCode") THEN
        SELECT '7:~Invalid Event code';
        RETURN
    END IF END IF;
CASE "pAction"
WHEN 'N' THEN
    INSERT INTO "contactevent"(
"contacteventid","staffid","personid","employmentid","vacancyid","contactdat
e","contacttime","callbackdate","callbacktime","description","classcode","no
tes","outcome","whoentered","divisionid","priority" ) VALUES(
"isnull"("pContactEventID","uniquekey"('x')), "pStaffID", "pPersonID", "pEmploy
mentID", "pVacancyID", "dContactDate", "dContactTime", "dCallbackDate", "dCallbac
kTime", "pDescription", "pClassCode", "pNotes", "pOutcome", "pStaffID", "pDivision
ID", "isnull"("pPriority",1) ) ;
    SELECT '0:~Success';
    RETURN
WHEN 'U' THEN -- Update record
    UPDATE "contactevent" SET "staffid" = "pStaffID", "personid" =
"isnull"("pPersonID", "personid"), "employmentid" =
"isnull"("pEmploymentID", "employmentid"), "vacancyid" =
"isnull"("pVacancyID", "vacancyid"), "description" =
"isnull"("pDescription", "description"), "classcode" =
"isnull"("pClassCode", "classcode"), "notes" =
```



```
"isnull"("pNotes","notes"),"outcome" =
"isnull"("pOutcome","outcome"),"priority" = "isnull"("pPriority","priority")
WHERE "ContactEventID" = "pContactEventID";
    IF "isnull"("dCallbackDate",'2000-01-01') <> '2000-01-01' THEN
        UPDATE "contactevent" SET "callbackdate" = "dCallbackDate" WHERE
"contacteventid" = "pContactEventID"
    END IF;
    IF "isnull"("dCallbackTime",'00:00:00.000') <> '00:00:00.000' THEN
        UPDATE "contactevent" SET "callbacktime" = "dCallbackTime" WHERE
"contacteventid" = "pContactEventID"
    END IF;
    SELECT '0:~Success';
    RETURN
WHEN 'D' THEN -- delete record
    DELETE FROM "contactevent" WHERE "contacteventid" = "pContactEventID";
    SELECT '0:~Success';
    RETURN
END CASE
END /* TEST
call NetTestSetup('');
delete from contactevent where contacteventid like 'TEST%';
insert into contactclass (classcode,classdescrip) on existing update values
('~','Test Class');
select * from
"NetOwnerContactEventsSet"('test.owner','TEST1','n','~','test','test','test
',dateformat(now(*),'dd/mm/yyyy
HH:nn:ss'),'TestDescription','TestNote',dateadd(hh,1,now(*)),'TestOutcome',1
);
expect pResult=0:~Success;
select * from contactevent where contacteventid='TEST1';
expect
personid=test,vacancyid=test,employmentid=test,description=TestDescription;
expect EOF;
select * from
"NetOwnerContactEventsSet"('test.owner','TEST2','Z',null,null,null,null,null
,null,null,null,null,null);
expect pResult=3:*;
select * from
"NetOwnerContactEventsSet"('test.owner',null,'U',null,null,null,null,null,nu
ll,null,null,null,null);
expect pResult=2:*;
select * from
"NetOwnerContactEventsSet"('test.owner','TEST2','N','~','test',null,null,'f
9',null,null,null,null,null);
expect pResult=4:*;
select * from
"NetOwnerContactEventsSet"('test.owner','TEST2','N',null,'test',null,null,da
teformat(now(*),'dd/mm/yyyy'),null,null,null,null,null);
```



```
expect pResult=8:*;
select * from
"NetOwnerContactEventsSet"('test.owner','TEST2','N','ZY','test',null,null,da
teformat(now(*),'dd/mm/yyyy'),null,null,null,null,null);
expect pResult=7:*;
select * from
"NetOwnerContactEventsSet"('test.owner','TEST1','U',null,null,null,null,null
,'TestDescription2',null,null,null,null);
expect pResult=0:~Success;
select * from
"NetOwnerContactEventsSet"('test.owner','TEST1','D',null,null,null,null,null
,'null',null,null,null,null);
expect pResult=0:~Success;
select * from contactevent where contacteventid='TEST1';
expect EOF;
select * from contactevent where contacteventid='TEST1';
expect EOF;
delete from contactevent where contacteventid like 'TEST%';
*/
/* DOC
2016-08-02 PC create,edit and delete contact events for Owner
2016-11-23 PC modify date handling
2018-08-27 PC improve error handler
```

Tests:

```
Create dummy contact class
Create new record
Create record with incorrect action
Create record with invalid contact event
Create record with event date
Create record with invalid staff
Create record with null event code
Create record with incorrect event code
modify existing record
delete record
*/
GO
```

```
COMMENT TO PRESERVE FORMAT ON PROCEDURE "pears"."NetOwnerContactEventsSet"
IS
{CREATE PROCEDURE pears."NetOwnerContactEventsSet"(IN pWebUserID CHAR(20),IN
pContactEventID CHAR(20) DEFAULT NULL,IN pAction CHAR(1),IN pClassCode
CHAR(5) DEFAULT NULL, IN pPersonID CHAR(20) DEFAULT NULL,IN pEmploymentID
CHAR(20) DEFAULT NULL,IN pVacancyID CHAR(20) DEFAULT NULL,IN pEventDate
CHAR(20) DEFAULT NULL,IN pDescription CHAR(100) DEFAULT NULL,IN pNotes long
VARCHAR DEFAULT NULL,IN pCallbackDate CHAR(20) DEFAULT NULL,IN pOutcome
CHAR(20) DEFAULT NULL,IN pPriority CHAR(1) DEFAULT NULL)
RESULT( "pResult" CHAR(250) )
```



```
BEGIN
  DECLARE dEventDate CHAR(20);
  DECLARE dContactDate CHAR(10);
  DECLARE dContactTime TIME;
  DECLARE pCallbackText CHAR(20);
  DECLARE dCallbackDate CHAR(10);
  DECLARE dCallbackTime TIME;
  DECLARE pStaffID CHAR(20);
  DECLARE pDivisionID CHAR(20);

  IF isnull(pEventDate, '') != '' THEN
    SET dEventDate=IQXNetStringToDate(LEFT(pEventDate,10));
    SET pEventDate=stuff(pEventDate,1,11,'');
    SET dContactTime=IQXNetStringToTime(pEventDate);
  ELSE
    SET dEventDate=IQXNetStringToDate(LEFT(now(*),10));
    SET dContactTime=NULL;
  END IF;
  SET dContactDate=dEventDate;

  IF dEventDate IS NULL AND pAction='N' THEN
    SELECT '4:~Invalid Event date';
    RETURN;
  END IF;

  IF isnull(pCallbackDate, '') != '' THEN
    SET dCallbackDate=IQXNetStringToDate(LEFT(pCallbackDate,10));
    SET pCallbackDate=stuff(pCallbackDate,1,11,'');
    SET dCallbackTime=IQXNetStringToTime(pCallbackDate);
  ELSE
    SET dCallbackDate=NULL;
    SET dCallbackTime=NULL;
  END IF;

  SET pStaffID=(SELECT staffid FROM iqxnetuser WHERE
iqxnetuserid=pWebUserID);
  SET pDivisionID=(SELECT divisionid FROM staff WHERE staffid=pStaffID);
  SET pCallbackText=(SELECT top 1 description FROM contactoutcome WHERE
contactoutcomeid=pOutcome);
  IF NOT pAction IN ('N','U','D') THEN
    SELECT '3:~Invalid Action';
    RETURN;
  END IF;
  IF isnull(pContactEventID, '')='' AND pAction IN ('U','D') THEN -- must
have a ContactEventID for update or delete
    SELECT '2:~pContactEventID required';
    RETURN;
  END IF;
```



```
IF pAction='N' THEN
  IF pClassCode IS NULL THEN
    SELECT '8:~Invalid Event code';
    RETURN;
  END IF;
  IF pClassCode NOT IN (SELECT classcode FROM contactclass WHERE
classcode=pClassCode) THEN
    SELECT '7:~Invalid Event code';
    RETURN;
  END IF;
END IF;
CASE pAction
  WHEN 'N' THEN
    INSERT INTO contactevent
(contacteventid,staffid,personid,employmentid,vacancyid,contactdate,contactt
ime,callbackdate,callbacktime,description,classcode,notes,outcome,whoentered
,divisionid,priority) VALUES
(isnull(pContactEventID,uniquekey('x')),pStaffID,pPersonID,pEmploymentID,pVa
cancyID,dContactDate,dContactTime,dCallbackDate,dCallbackTime,pDescription,p
ClassCode,pNotes,pOutcome,pStaffID,pDivisionID,isnull(pPriority,1));
    SELECT '0:~Success';
    RETURN;
  WHEN 'U' THEN -- Update record
    UPDATE contactevent SET staffid=pStaffID,
personid=isnull(pPersonID,personid),
employmentid=isnull(pEmploymentID,employmentid),
vacancyid=isnull(pVacancyID,vacancyid),
description=isnull(pDescription,description),
classcode=isnull(pClassCode,classcode),notes=isnull(pNotes,notes),outcome=is
null(pOutcome,outcome),priority=isnull(pPriority,priority) WHERE
ContactEventID=pContactEventID;
    IF isnull(dCallbackDate,'2000-01-01') !='2000-01-01' THEN
      UPDATE contactevent SET callbackdate=dCallbackDate WHERE
contacteventid=pContactEventID;
    END IF;
    IF isnull(dCallbackTime,'00:00:00.000') !='00:00:00.000' THEN
      UPDATE contactevent SET callbacktime=dCallbackTime WHERE
contacteventid=pContactEventID;
    END IF;
    SELECT '0:~Success';
    RETURN;
  WHEN 'D' THEN -- delete record
    DELETE FROM contactevent WHERE contacteventid=pContactEventID;
    SELECT '0:~Success';
    RETURN;
END CASE;
END
/* TEST
```



```
call NetTestSetup('');
delete from contactevent where contacteventid like 'TEST%';
insert into contactclass (classcode,classdescrip) on existing update values
('~','Test Class');
select * from
"NetOwnerContactEventsSet"('test.owner','TEST1','n','~','test','test','test
',dateformat(now(*),'dd/mm/yyyy
HH:nn:ss'),'TestDescription','TestNote',dateadd(hh,1,now(*)),'TestOutcome',1
);
expect pResult=0:~Success;
select * from contactevent where contacteventid='TEST1';
expect
personid=test,vacancyid=test,employmentid=test,description=TestDescription;
expect EOF;
select * from
"NetOwnerContactEventsSet"('test.owner','TEST2','Z',null,null,null,null,null
,null,null,null,null,null);
expect pResult=3:*;
select * from
"NetOwnerContactEventsSet"('test.owner',null,'U',null,null,null,null,null,nu
ll,null,null,null,null);
expect pResult=2:*;
select * from
"NetOwnerContactEventsSet"('test.owner','TEST2','N','~','test',null,null,'f
9',null,null,null,null,null);
expect pResult=4:*;
select * from
"NetOwnerContactEventsSet"('test.owner','TEST2','N',null,'test',null,null,da
teformat(now(*),'dd/mm/yyyy'),null,null,null,null,null);
expect pResult=8:*;
select * from
"NetOwnerContactEventsSet"('test.owner','TEST2','N','ZY','test',null,null,da
teformat(now(*),'dd/mm/yyyy'),null,null,null,null,null);
expect pResult=7:*;
select * from
"NetOwnerContactEventsSet"('test.owner','TEST1','U',null,null,null,null,null
,'TestDescription2',null,null,null,null);
expect pResult=0:~Success;
select * from
"NetOwnerContactEventsSet"('test.owner','TEST1','D',null,null,null,null,null
,'null',null,null,null,null);
expect pResult=0:~Success;
select * from contactevent where contacteventid='TEST1';
expect EOF;
select * from contactevent where contacteventid='TEST1';
expect EOF;
delete from contactevent where contacteventid like 'TEST%';
*/
```



```
/* DOC
2016-08-02 PC create,edit and delete contact events for Owner
2016-11-23 PC modify date handling
2018-08-27 PC improve error handler
```

Tests:

```
    Create dummy contact class
    Create new record
    Create record with incorrect action
    Create record with invalid contact event
    Create record with event date
    Create record with invalid staff
    Create record with null event code
    Create record with incorrect event code
    modify existing record
    delete record
```

```
*/
}
```

From:

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

Permanent link:

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

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

