



pears.NetOwnerRequirementShortList



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

Original SQL

```
CREATE PROCEDURE "pears"."NetOwnerRequirementShortList"( IN "pWebUserID"
CHAR(20),IN "pRowID" CHAR(20),IN "IsShift" CHAR(1) )
RESULT( "PersonName" CHAR(60),"PersonID" CHAR(20),"PersonStatus"
CHAR(50),"Requested" CHAR(1),"Rejected" CHAR(1),"WithHeld" CHAR(1) )
// IQXWeb
BEGIN
  DECLARE "VacID" CHAR(20);
  IF "IsShift" = 0 THEN
    SELECT "pe"."name",
           "pe"."personid",
           "s"."name",
           (SELECT "count"() FROM "progress" WHERE "status" = 'P' AND "personid"
= "pe"."personid" AND "vacancyid" = "p"."vacancyid") AS "Requested",
           (SELECT "count"() FROM "progress" WHERE "status" = 'R' AND "personid"
= "pe"."personid" AND "vacancyid" = "p"."vacancyid") AS "Rejected",
           (SELECT "count"() FROM "withholds" AS "w" WHERE "w"."personid" =
"p"."personid" AND("p"."vacancyid" = "w"."vacancyid" OR "e"."companyid" =
"w"."companyid")) AS "WithHeld"
    FROM "employment" AS "e"
        KEY JOIN "vacancy"
        KEY JOIN "progress" AS "p"
        KEY JOIN "person" AS "pe"
        , "progress" AS "p"
        JOIN "status" AS "s" ON "s"."status" = "p"."status" AND "s"."type" =
'R'
    WHERE "p"."vacancyid" = "pRowID"
    AND "s"."name" NOT LIKE 'placed%'
    AND "p"."rejectionreason" IS NULL
    AND "WithHeld" = 0
  ELSE
    SET "VacID" = (SELECT "VacancyID" FROM "TempShiftPlan" WHERE
"tempshiftplanid" = "pRowID");
    SELECT "pe"."name",
           "pe"."personid",
           "s"."name",
```



```
(SELECT "count"() FROM "tempshiftprogress" WHERE "status" = 'P' AND
"personid" = "pe"."personid" AND "tempshiftplanid" = "n"."tempshiftplanid")
AS "Requested",
(SELECT "count"() FROM "tempshiftprogress" WHERE "status" = 'R' AND
"personid" = "pe"."personid" AND "tempshiftplanid" = "n"."tempshiftplanid")
AS "Rejected",
(SELECT "count"() FROM "withholds" AS "w" WHERE "w"."personid" =
"p"."personid" AND("pl"."vacancyid" = "w"."vacancyid" OR "e"."companyid" =
"w"."companyid")) AS "WithHeld"
FROM "employment" AS "e"
KEY JOIN "vacancy"
KEY JOIN "tempshiftplan" AS "pl"
KEY JOIN "tempshiftprogress" AS "p"
KEY JOIN "person" AS "pe"
,"tempshiftprogress" AS "p"
JOIN "status" AS "s" ON "s"."status" = "p"."status" AND "s"."type" =
'R'
,"tempshiftprogress" AS "p"
KEY JOIN "tempshiftplan" AS "n"
WHERE "p"."tempshiftplanid" = "pRowID"
AND "s"."name" <> 'placed'
AND "p"."rejectionreason" IS NULL
AND "WithHeld" = 0
END IF
END /* DOC
```

This routine requires contact event types CI and CD

2016-11-30 PC add requested / rejected, and add doc and test

2017-11-17 PC IW-266 remove candidate if Withheld

2018-01-03 PC IW-300,35 bug fix on withheld

2018-05-04 PC IW-482 it is now possible to view vacancies other than your own so remove owner filters

Tests:

invalid pWebUserID

valid pWebUserID, invalid Rowid

valid pWebUserID, Rowid invalid IsShift

valid pWebUserID, Rowid, IsShift=0 with no events

valid pWebUserID, Rowid, IsShift=0 with requested

valid pWebUserID, Rowid, IsShift=0 with rejected

valid pWebUserID, Rowid, IsShift=1 with requested and rejected

**/*

/ 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.owner';
insert into progress (progressid,vacancyid,personid,status,staffid) on
existing skip values ('TEST','TEST','TEST','#','TEST');
update vacancy set staffid='TEST' where vacancyid='TEST';
select * from NetOwnerRequirementShortList(null,null,null);
expect EOF;
select * from NetOwnerRequirementShortList('test.owner',null,null);
expect EOF;
select * from NetOwnerRequirementShortList('test.owner','zz',null);
expect EOF;
select * from NetOwnerRequirementShortList('test.owner','TEST',null) order
by PersonStatus asc;
expect PersonName=test;
select * from NetOwnerRequirementShortList('test.owner','TEST',0) order by
PersonStatus asc;
expect PersonName=test;
insert into progress (progressid,vacancyid,personid,status,staffid) on
existing skip values ('TEST2','TEST','TEST','P','TEST');
select * from NetOwnerRequirementShortList('test.owner','TEST',0);
expect PersonName=test,PersonID=TEST,PersonStatus=*,Requested=1,Rejected=0;
insert into progress (progressid,vacancyid,personid,status,staffid) values
('TEST3','TEST','TEST','R','TEST');
select * from NetOwnerRequirementShortList('test.owner','TEST',0);
expect PersonName=test,PersonID=TEST,PersonStatus=*,Requested=1,Rejected=1;
insert into tempshiftprogress
(tempshiftprogressid,tempshiftplanid,status,staffid,personid)
values('TEST','TEST','A','TEST','TEST');
select * from NetOwnerRequirementShortList('test.owner','TEST',1);
expect PersonName=test,PersonID=TEST,PersonStatus=*,Requested=0,Rejected=0;
insert into withholds (WithHoldsID,PersonID,CompanyID) values
('TEST','TEST','TEST');
select * from NetOwnerRequirementShortList('test.client','TEST',0);
expect EOF;
update withholds set CompanyID=null,VacancyID='TEST' where
WithHoldsID='TEST';
select * from NetOwnerRequirementShortList('test.client','TEST',0);
expect EOF;
delete from withholds where WithHoldsID='TEST';

*/
GO
```

```
COMMENT TO PRESERVE FORMAT ON PROCEDURE
"pears"."NetOwnerRequirementShortList" IS
{CREATE PROCEDURE pears."NetOwnerRequirementShortList"(IN pWebUserID
CHAR(20),IN pRowID CHAR(20),IN IsShift CHAR(1))
RESULT(PersonName CHAR(60),PersonID CHAR(20),PersonStatus CHAR(50),Requested
```



```
CHAR(1),Rejected CHAR(1),WithHeld CHAR(1))
// IQXWeb
BEGIN
  DECLARE VacID CHAR(20);
  IF IsShift=0 THEN
    SELECT
      pe.name,
      pe.personid,
      s.name,
      (SELECT COUNT(*) FROM progress WHERE STATUS = 'P' AND
personid=pe.personid AND vacancyid=p.vacancyid) AS Requested,
      (SELECT COUNT(*) FROM progress WHERE STATUS = 'R' AND
personid=pe.personid AND vacancyid=p.vacancyid) AS Rejected,
      (SELECT COUNT(*) FROM withholds w WHERE w.personid=p.personid
AND (p.vacancyid=w.vacancyid OR e.companyid=w.companyid)) AS WithHeld
    FROM
      employment e
      KEY JOIN vacancy
      KEY JOIN progress p
      KEY JOIN person pe,
      progress p
      JOIN STATUS s ON s.status=p.status AND s.type='R'
  WHERE
    p.vacancyid=pRowID
    AND s.name NOT LIKE 'placed%'
    AND p.rejectionreason IS NULL
    AND WithHeld=0
  ELSE
    SET "VacID" = (SELECT VacancyID FROM "TempShiftPlan" WHERE
"tempshiftplanid" = pRowID);
    SELECT
      pe.name,
      pe.personid,
      s.name,
      (SELECT COUNT(*) FROM tempshiftprogress WHERE STATUS = 'P' AND
personid=pe.personid AND tempshiftplanid=n.tempshiftplanid) AS Requested,
      (SELECT COUNT(*) FROM tempshiftprogress WHERE STATUS = 'R' AND
personid=pe.personid AND tempshiftplanid=n.tempshiftplanid) AS Rejected,
      (SELECT COUNT(*) FROM withholds w WHERE w.personid=p.personid
AND (pl.vacancyid=w.vacancyid OR e.companyid=w.companyid)) AS WithHeld
    FROM
      employment e
      KEY JOIN vacancy
      KEY JOIN tempshiftplan pl
      KEY JOIN tempshiftprogress p
      KEY JOIN person pe,
      tempshiftprogress p
      JOIN STATUS s ON s.status=p.status AND s.type='R',
```



```
tempshiftprogress p
KEY JOIN tempshiftplan n
WHERE
  p.tempshiftplanid=pRowID
  AND s.name !='placed'
  AND p.rejectionreason IS NULL
  AND WithHeld=0;
END IF;
END
```

```
/* DOC
```

This routine requires contact event types CI and CD

2016-11-30 PC add requested / rejected, and add doc and test

2017-11-17 PC IW-266 remove candidate if Withheld

2018-01-03 PC IW-300,35 bug fix on withheld

2018-05-04 PC IW-482 it is now possible to view vacancies other than your own so remove owner filters

Tests:

```
invalid pWebUserID
valid pWebUserID, invalid Rowid
valid pWebUserID, Rowid invalid IsShift
valid pWebUserID, Rowid, IsShift=0 with no events
valid pWebUserID, Rowid, IsShift=0 with requested
valid pWebUserID, Rowid, IsShift=0 with rejected
valid pWebUserID, Rowid, IsShift=1 with requested and rejected
```

```
*/
```

```
/* 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.owner';
insert into progress (progressid,vacancyid,personid,status,staffid) on
existing skip values ('TEST','TEST','TEST','#','TEST');
update vacancy set staffid='TEST' where vacancyid='TEST';
select * from NetOwnerRequirementShortList(null,null,null);
expect EOF;
select * from NetOwnerRequirementShortList('test.owner',null,null);
expect EOF;
select * from NetOwnerRequirementShortList('test.owner','zz',null);
expect EOF;
select * from NetOwnerRequirementShortList('test.owner','TEST',null) order
by PersonStatus asc;
```



```
expect PersonName=test;
select * from NetOwnerRequirementShortList('test.owner','TEST',0) order by
PersonStatus asc;
expect PersonName=test;
insert into progress (progressid,vacancyid,personid,status,staffid) on
existing skip values ('TEST2','TEST','TEST','P','TEST');
select * from NetOwnerRequirementShortList('test.owner','TEST',0);
expect PersonName=test,PersonID=TEST,PersonStatus=*,Requested=1,Rejected=0;
insert into progress (progressid,vacancyid,personid,status,staffid) values
('TEST3','TEST','TEST','R','TEST');
select * from NetOwnerRequirementShortList('test.owner','TEST',0);
expect PersonName=test,PersonID=TEST,PersonStatus=*,Requested=1,Rejected=1;
insert into tempshiftprogress
(tempshiftprogressid,tempshiftplanid,status,staffid,personid)
values('TEST','TEST','A','TEST','TEST');
select * from NetOwnerRequirementShortList('test.owner','TEST',1);
expect PersonName=test,PersonID=TEST,PersonStatus=*,Requested=0,Rejected=0;
insert into withholds (WithHoldsID,PersonID,CompanyID) values
('TEST','TEST','TEST');
select * from NetOwnerRequirementShortList('test.client','TEST',0);
expect EOF;
update withholds set CompanyID=null,VacancyID='TEST' where
WithHoldsID='TEST';
select * from NetOwnerRequirementShortList('test.client','TEST',0);
expect EOF;
delete from withholds where WithHoldsID='TEST';

*/
}
```

From:

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

Permanent link:

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

Last update: 2026/08/07 19:24

