



# pears.NetOwnerVacancySuggestCandidate



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

## Original SQL

```
CREATE PROCEDURE "pears"."NetOwnerVacancySuggestCandidate"( IN "pWebUserID"
CHAR(20),IN "pVacancyID" CHAR(20),IN "pSlice" INTEGER DEFAULT 0,IN
"pSliceSize" INTEGER DEFAULT 10,IN "pSearchString" CHAR(50) )
RESULT( "pPersonID" CHAR(20),"pName" CHAR(60),"pKeyName" CHAR(60) )
// IQXNet
BEGIN
    DECLARE LOCAL TEMPORARY TABLE "PersonAlreadyWorking"(
        "PersonID" CHAR(20) NULL,
    ) NOT transactional;
    DECLARE "x" INTEGER;
    DECLARE "y" INTEGER;
    DECLARE "CompID" CHAR(20);
    DECLARE "VacID" CHAR(20);
    DECLARE "SearchString" CHAR(50);
    IF "pSearchString" = 'undefined' THEN
        SET "SearchString" = '%'
    ELSE
        SET "SearchString" = "pSearchString"
    END IF;
    SET "x" = "pSliceSize";
    SET "y" = "x"*"pSlice"+1;
    SET "CompID" = "isnull"((SELECT "companyid" FROM "employment" KEY JOIN
"vacancy" WHERE "vacancyid" = "pVacancyID"),(SELECT "companyid" FROM
"employment" KEY JOIN "vacancy" KEY JOIN "tempshiftplan" WHERE
"tempshiftplanid" = "pVacancyID"));
    SET "VacID" = "isnull"((SELECT top 1 "vacancyid" FROM "tempshiftplan"
WHERE "tempshiftplanid" = "pVacancyID"),"pVacancyID");
    INSERT INTO "PersonAlreadyWorking"( "PersonID" ) (SELECT DISTINCT
        "ts"."personid"
    FROM "tempshiftplan" AS "tsp"
        JOIN "tempshift" AS "ts" ON "ts"."shiftdate" = "tsp"."shiftdate" AND
"ts"."state" <> 'A' AND NOT(("isnull"("ts"."timeto",'23:59:00') <=
"tsp"."timefrom") OR("isnull"("ts"."timefrom",'00:00:00') >=
"tsp"."timeto"))
        JOIN "person" AS "p"
```



```
WHERE "ts"."vacancyid" = "VacID"
AND "p"."status" IN( 'C' );
SELECT DISTINCT top "x" START at "y"
  "p"."personid", "p"."name", "p"."keyname"
FROM "employment" AS "e"
  KEY JOIN "vacancy" AS "v"
  KEY JOIN "placement" AS "pl"
  KEY JOIN "employment"
  KEY JOIN "person" AS "p"
  LEFT OUTER JOIN "PersonAlreadyWorking" AS "paw" ON "paw"."PersonID" =
  "p"."personid"
WHERE "paw"."personID" IS NULL
AND "p"."status" IN( 'C' )
AND "e"."companyid" = "CompID"
AND "p"."keyname" LIKE '%'+ "SearchString" + '%'
AND (SELECT "IQXNetUserClassID" FROM "IQXNetUser" WHERE "IQXNetUserID" =
  "pWebUserID") = 'OWNER'
ORDER BY "p"."keyname" ASC
END /* DOC
```

*Used on Owner Requirements page*

*2016-12-28 PC provide paged list of candidates who would be available to work in a particular vacancy*

*2017-05-17 PC default search adjusted*

*2018-07-30 PC extend schema to include kayname*

*2018-09-28 ET Added prefixes to all fields without them.*

*2018-09-28 ET removed iqxnetuserlink stuff from test*

*Tests:*

*unable to test invalid slice or sliceSize*

*invalid pWebUserID*

*valid pWebUserID, invalid pVacancyID, invalid status*

*valid pWebUserID, pVacancyID, pSlice, pSliceSize, invalid pSearchString*

*valid pWebUserID, pVacancyID, pSlice, pSliceSize, pSearchString*

*valid pWebUserID, pVacancyID, pSlice, pSliceSize, pSearchString with shift for today and time wholly before template times*

*valid pWebUserID, pVacancyID, pSlice, pSliceSize, pSearchString with shift for today and time wholly overlapping template times*

*valid pWebUserID, pVacancyID, pSlice, pSliceSize, pSearchString with shift for today and time wholly after template times*

*valid pWebUserID, pVacancyID, pSlice, pSliceSize, pSearchString with shift for tomorrow*

*\*/*

*/\* TEST*

*call NetTestSetup('');*

*update tempshiftplan set shiftdate=today(\*) where tempshiftplanid = 'TEST';*

*update tempshift set personid=null where tempshiftid='TEST';*

*update person set status='~',keyname='aaatest' where personid='TEST';*



```
select * from NetOwnerVacancySuggestCandidate(null,null,0,1,null) where
pPersonID='TEST';
expect EOF;
update person set status='C' where personid='TEST';
select * from NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,null)
where pPersonID='TEST';
expect pPersonID=TEST;
select * from
NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,'test');
expect pPersonID=TEST;
update tempshift set personid='TEST' where tempshiftid='TEST';
select * from NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,null)
where pPersonID='TEST';
expect pPersonID=TEST;
update tempshiftplan set timefrom='10:00',timeto='12:00' where
tempshiftplanid='TEST';
update tempshift set timefrom='07:00',timeto='09:00' where tempshiftid =
'TEST';
select * from
NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,'test');
expect pPersonID=TEST;
update tempshift set timefrom='07:00',timeto='15:00' where tempshiftid =
'TEST';
select * from
NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,'test') where
ppersonid='TEST';
expect EOF;
update tempshift set timefrom='14:00',timeto='15:00' where tempshiftid =
'TEST';
select * from
NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,'test');
expect pPersonID=TEST;
update tempshiftplan set shiftdate=dateadd(day,1,today(*)) where
tempshiftplanid = 'TEST';
select * from
NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,'test');
expect pPersonID=TEST;
update person set status=null,keyname='test' where personid='TEST';
update tempshiftplan set shiftdate=null where tempshiftplanid = 'TEST';
*/
GO
```

COMMENT TO PRESERVE FORMAT ON PROCEDURE

```
"pears"."NetOwnerVacancySuggestCandidate" IS
{CREATE PROCEDURE pears."NetOwnerVacancySuggestCandidate"(IN pWebUserID
CHAR(20), IN pVacancyID CHAR(20), IN pSlice INTEGER DEFAULT 0, IN pSliceSize
INTEGER DEFAULT 10, IN pSearchString CHAR(50))
RESULT(pPersonID CHAR(20), pName CHAR(60),pKeyName CHAR(60))
```



```
// IQXNet
BEGIN
  DECLARE LOCAL TEMPORARY TABLE PersonAlreadyWorking(
    PersonID CHAR(20) NULL,
  ) NOT transactional;
  DECLARE x INTEGER;
  DECLARE y INTEGER;
  DECLARE CompID CHAR(20);
  DECLARE VacID CHAR(20);
  DECLARE SearchString CHAR(50);

  IF pSearchString='undefined' THEN
    SET SearchString='%'
  ELSE
    SET SearchString=pSearchString
  END IF;

  SET x=pSliceSize;
  SET y=x*pSlice+1;
  SET CompID = isnull((SELECT companyid FROM employment KEY JOIN vacancy
WHERE vacancyid=pVacancyID),(SELECT companyid FROM employment KEY JOIN
vacancy KEY JOIN tempshiftplan WHERE tempshiftplanid = pVacancyID ));
  SET VacID=isnull((SELECT top 1 vacancyid FROM tempshiftplan WHERE
tempshiftplanid = pVacancyID),pVacancyID);
  INSERT INTO PersonAlreadyWorking (PersonID) (SELECT DISTINCT
    ts.personid
  FROM
    tempshiftplan tsp
    JOIN tempshift ts ON ts.shiftdate = tsp.shiftdate AND ts.state !=
'A' AND NOT ((isnull(ts.timeto,'23:59:00') <= tsp.timefrom) OR
(isnull(ts.timefrom,'00:00:00') >= tsp.timeto))
    JOIN person p
    WHERE ts.vacancyid = VacID
    AND p.status IN ('C')
  );
  SELECT DISTINCT top x START at y
    p.personid,p.name,p.keyname
  FROM
    employment e
    KEY JOIN vacancy v
    KEY JOIN placement pl
    KEY JOIN employment
    KEY JOIN person p
    LEFT OUTER JOIN PersonAlreadyWorking paw ON paw.PersonID=p.personid
  WHERE
    paw.personID IS NULL
    AND p.status IN ('C')
    AND e.companyid = CompID
```



```
        AND p.keyname LIKE '%' + SearchString + '%'
        AND (SELECT IQXNetUserClassID FROM IQXNetUser WHERE
IQXNetUserID=pWebUserID)='OWNER'
    ORDER BY
        p.keyname ASC;
END
```

```
/* DOC
```

```
Used on Owner Requirements page
```

2016-12-28 PC provide paged list of candidates who would be available to work in a particular vacancy

2017-05-17 PC default search adjusted

2018-07-30 PC extend schema to include kayname

2018-09-28 ET Added prefixes to all fields without them.

2018-09-28 ET removed iqxnetuserlink stuff from test

Tests:

unable to test invalid slice or sliceSize

invalid pWebUserID

valid pWebUserID, invalid pVacancyID, invalid status

valid pWebUserID, pVacancyID, pSlice, pSliceSize, invalid pSearchString

valid pWebUserID, pVacancyID, pSlice, pSliceSize, pSearchString

valid pWebUserID, pVacancyID, pSlice, pSliceSize, pSearchString with

shift for today and time wholly before template times

valid pWebUserID, pVacancyID, pSlice, pSliceSize, pSearchString with

shift for today and time wholly overlapping template times

valid pWebUserID, pVacancyID, pSlice, pSliceSize, pSearchString with

shift for today and time wholly after template times

valid pWebUserID, pVacancyID, pSlice, pSliceSize, pSearchString with

shift for tomorrow

\*/

```
/* TEST
```

```
call NetTestSetup('');
```

```
update tempshiftplan set shiftdate=today(*) where tempshiftplanid = 'TEST';
```

```
update tempshift set personid=null where tempshiftid='TEST';
```

```
update person set status='~',keyname='aaatest' where personid='TEST';
```

```
select * from NetOwnerVacancySuggestCandidate(null,null,0,1,null) where
pPersonID='TEST';
```

```
expect EOF;
```

```
update person set status='C' where personid='TEST';
```

```
select * from NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,null)
where pPersonID='TEST';
```

```
expect pPersonID=TEST;
```

```
select * from
```

```
NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,'test');
```

```
expect pPersonID=TEST;
```



```
update tempshift set personid='TEST' where tempshiftid='TEST';
select * from NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,null)
where pPersonID='TEST';
expect pPersonID=TEST;
update tempshiftplan set timefrom='10:00',timeto='12:00' where
tempshiftplanid='TEST';
update tempshift set timefrom='07:00',timeto='09:00' where tempshiftid =
'TEST';
select * from
NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,'test');
expect pPersonID=TEST;
update tempshift set timefrom='07:00',timeto='15:00' where tempshiftid =
'TEST';
select * from
NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,'test') where
ppersonid='TEST';
expect EOF;
update tempshift set timefrom='14:00',timeto='15:00' where tempshiftid =
'TEST';
select * from
NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,'test');
expect pPersonID=TEST;
update tempshiftplan set shiftdate=dateadd(day,1,today(*)) where
tempshiftplanid = 'TEST';
select * from
NetOwnerVacancySuggestCandidate('test.owner','TEST',0,1,'test');
expect pPersonID=TEST;
update person set status=null,keyname='test' where personid='TEST';
update tempshiftplan set shiftdate=null where tempshiftplanid = 'TEST';
*/
}
```

From:

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

Permanent link:

[https://iqxusers.co.uk/iqxhelp/doku.php?id=database:procedures:pears\\_netownervacancysuggestcandidate](https://iqxusers.co.uk/iqxhelp/doku.php?id=database:procedures:pears_netownervacancysuggestcandidate)

Last update: 2026/08/07 19:24

