



pears.NetClientAddVacancies



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

Original SQL

```
CREATE PROCEDURE "pears"."NetClientAddVacancies"( IN "pWebUserID"
CHAR(20),IN "ptemplatevacancyid" CHAR(50),IN "pdatefrom" CHAR(25),
  IN "pdateto" CHAR(25) DEFAULT NULL,IN "pTheirRef" CHAR(50) DEFAULT NULL,IN
"pTitle" CHAR(100) DEFAULT NULL,IN "pDescription" long VARCHAR DEFAULT
NULL,IN "pquantity" CHAR(25) DEFAULT '1' )
RESULT( "pResult" CHAR(250) )
// IQXWeb
BEGIN
  DECLARE "swebusername" CHAR(50);
  DECLARE "sid" CHAR(20);
  DECLARE "cnt" INTEGER;
  DECLARE "dfrom" DATE;
  DECLARE "dto" DATE;
  DECLARE "CanAuthorise" SMALLINT;
  DECLARE "NewVacancyState" CHAR(1);
  DECLARE LOCAL TEMPORARY TABLE "IDs"(
    "ID" CHAR(20) NULL,
  ) NOT transactional;
  INSERT INTO "IDs"( "ID" )
    SELECT "companyid" FROM "employment" KEY JOIN "iqxnetuserlink" WHERE
"iqxnetuserid" = "pWebUserID";
  IF NOT "ptemplatevacancyid" = any(SELECT "v"."vacancyid" FROM "vacancy" AS
"v" KEY JOIN "employment" AS "e" JOIN "IDs" ON "e"."companyid" = "IDs"."ID")
THEN
    SELECT '99:~Permission denied';
    RETURN
  END IF;
  IF "IQXNetHasPermission"("pWebUserID",'ORDERJOBS') = 0 THEN
    SELECT '99:~Permission denied';
    RETURN
  END IF;
  SET "swebusername" = (SELECT "name" FROM "iqxnetuser" WHERE "iqxnetuserid"
= "pwebuserid");
  SET "cnt" = "iqxnetstringtointeger"("pquantity");
  IF "cnt" IS NULL OR "cnt" = 0 THEN
```



```
SELECT '450:~Invalid quantity';
RETURN
END IF;
SET "dfrom" = "iqxnetstringtodate"("pdatefrom");
IF "dfrom" IS NULL THEN
    SELECT '500:~Invalid start date';
    RETURN
END IF;
SET "pdateto" = "nullif"("trim"("pdateto"), '');
SET "dto" = "iqxnetstringtodate"("pdateto");
IF "pdateto" IS NOT NULL AND "dto" IS NULL THEN
    SELECT '500:~Invalid end date';
    RETURN
END IF;
SET "CanAuthorise" = "IQXNetHasPermission"("pWebUserID", 'AUTHORISEJOBS');
IF "CanAuthorise" = 1 THEN
    SET "NewVacancyState" = 'C'
ELSE
    SET "NewVacancyState" = ']'
END IF;
SET "sid" = "uniquekey"("cnt");
INSERT INTO "vacancy"(
"vacancyid", "departmentid", "employmentid", "staffid", "entrydate", "status", "startdate", "FinishDate", "temp", "position", "notes", "othernotes", "classcode", "ErNI", "HolidayAllowance", "Discount",
"TempDeskID", "PayrollIdentifier", "AnalysisCode", "clientdepartment", "Currency", "RecipientID", "Role", "SiteName", "SiteContact", "addr1", "addr2", "addr3", "town", "county", "country", "postcode", "TempJobTypeID",
"SitePhoneNumbers", "Invaddr1", "Invaddr2", "Invaddr3", "Invtown", "Invcounty", "Invcountry", "Invpostcode", "SiteFax", "SiteEmail", "TheirRef",
"PriceListID", "SendShiftsToPayroll", "WorkMonday", "WorkTuesday", "WorkWednesday", "WorkThursday", "WorkFriday", "WorkSaturday", "WorkSunday", "WorkNormalHours", "WorkStartTime", "noofposts" )
    SELECT "sid", "departmentid", "employmentid", "staffid", CURRENT
DATE, "NewVacancyState", "dfrom", "dto", "temp", "isnull"("pTitle", "position"), "notes", "isnull"("pDescription", "othernotes"), "classcode", "ErNI", "HolidayAllowance", "Discount",
"TempDeskID", "PayrollIdentifier", "AnalysisCode", "clientdepartment", "Currency", "RecipientID", "Role", "SiteName", "SiteContact", "addr1", "addr2", "addr3", "town", "county", "country", "postcode", "TempJobTypeID",
"SitePhoneNumbers", "Invaddr1", "Invaddr2", "Invaddr3", "Invtown", "Invcounty", "Invcountry", "Invpostcode", "SiteFax", "SiteEmail", "ucase"("pTheirRef"),
"PriceListID", "SendShiftsToPayroll", "WorkMonday", "WorkTuesday", "WorkWednesday", "WorkThursday", "WorkFriday", "WorkSaturday", "WorkSunday", "WorkNormalHours", "WorkStartTime", "cnt"
    FROM "vacancy" WHERE "vacancyid" = "ptemplatevacancyid";
INSERT INTO "tempjobrate"(
"TempJobRateID", "VacancyID", "TempPayBandID", "PayRate", "ChargeRate", "StartDat
```



```
e", "EndDate", "Grade" )
SELECT
"uniquekey"("TempJobRateID"), "sid", "TempPayBandID", "PayRate", "ChargeRate", "S
tartDate", "EndDate", "Grade"
FROM "tempjobrate" WHERE "vacancyid" = "ptemplatevacancyid";
INSERT INTO "criterion"(
"searchlocation", "id", "critid", "sourcelocation", "dictionaryid", "tagid", "tagc
hoiceid", "notflag", "matchtype", "textvalue", "value", "uppervalue", "taganytotal
", "extravalue", "extrauppervalue", "extramatchtype" )
SELECT
"searchlocation", "sid", "critid", "sourcelocation", "dictionaryid", "tagid", "tag
choiceid", "notflag", "matchtype", "textvalue", "value", "uppervalue", "taganytota
l", "extravalue", "extrauppervalue", "extramatchtype"
FROM "criterion" WHERE "searchlocation" LIKE 'V%' AND "id" =
"ptemplatevacancyid";
CALL "NetSetVacancyCascade"("sid");
IF "CanAuthorise" = 1 THEN
CALL
"IQXNetPopup"("pWebUserID", NULL, NULL, "sid", NULL, NULL, 'ADDVACANCY', 'New
vacancy requirement added', NULL)
END IF;
SELECT '0:~Success'
END /* DOC
2016-10-25 PC IW-69 test and doc sections added
2018-08-22 PC improve test handling
2019-02-20 PC IW-838 create only one vacancy with multiple count, instead of
looping to create multiple vacancies

tests:
invalid pWebUserID
valid pWebUserID and invalid pTemplateVacancyID
valid pWebUserID, valid pTemplateVacancyID which does not belong to
pWebUserID
valid pWebUserID, valid pTemplateVacancyID but ORDERJOBS set false
valid pWebUserID, valid pTemplateVacancyID, ORDERJOBS set true but invalid
quantity
valid pWebUserID, valid pTemplateVacancyID, ORDERJOBS set true, valid
quantity but invalid start date
valid pWebUserID, valid pTemplateVacancyID, ORDERJOBS set true, valid
quantity, valid start date but invalid end date
valid pWebUserID, valid pTemplateVacancyID, ORDERJOBS set true, valid
quantity, valid start date and null end date
valid pWebUserID, valid pTemplateVacancyID, ORDERJOBS set true, valid
quantity, valid start date and null valid date
all fields valid and AUTHORISEJOBS set to 0
all fields valid and AUTHORISEJOBS set to 1

*/
```



```
/* 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.client';
update vacancy set status='[' where vacancyid='test';
delete from iqxnetrightallowed where iqxnetrightid='ORDERJOBS' and
iqxnetuserclassid='client';
delete from iqxnetrightallowed where iqxnetrightid='AUTHORISEJOBS' and
iqxnetuserclassid='client';
select * from
NetClientAddVacancies(null,null,null,null,null,null,null,null);
expect pResult=99*;
select * from
NetClientAddVacancies('test.x',null,null,null,null,null,null,null);
expect pResult=99*;
select * from
NetClientAddVacancies('test.client',null,null,null,null,null,null,null);
expect pResult=99*;
insert into iqxnetrightallowed values ('ORDERJOBS','CLIENT','*',1);
select * from
NetClientAddVacancies('test.client','test',null,null,null,null,null,null);
expect pResult=450*;
select * from
NetClientAddVacancies('test.client','test',null,null,null,null,null,1);
expect pResult=500*;
select * from
NetClientAddVacancies('test.client','test',dateformat(now(*),'dd/mm/yyyy'),'
xx',null,null,null,1);
expect pResult=500*;
select * from
NetClientAddVacancies('test.client','test',dateformat(now(*),'dd/mm/yyyy'),n
ull,null,null,null,1);
expect pResult=0*;
select top 1 departmentid,employmentid,(if
entrydate=dateformat(now(*),'yyyy-mm-dd') then 1 else 0 endif) as
A,status,position,(if startdate=dateformat(now(*),'yyyy-mm-dd') then 1 else
0 endif) as B,(if
isnull(finishdate,'1900-01-01')=isnull(dateformat(dateadd(dd,2,now(*)),'yyyy
-mm-dd'),'1900-01-01') then 1 else 0 endif) as C from vacancy order by
whenentered desc;
expect departmentid=~~,employmentid=TEST,A=1,status=],position=test,B=1,C=0;
select * from
NetClientAddVacancies('test.client','test',dateformat(now(*),'dd/mm/yyyy'),d
ateformat(dateadd(dd,2,now(*)),'dd/mm/yyyy'),null,null,null,1);
expect pResult=0*;
select top 1 departmentid,employmentid,(if
```



```
entrydate=dateformat(now(*),'yyyy-mm-dd') then 1 else 0 endif) as
D,status,position,(if startdate=dateformat(now(*),'yyyy-mm-dd') then 1 else
0 endif) as E,(if
isnull(finishdate,'1900-01-01')=isnull(dateformat(dateadd(dd,2,now(*)),'yyyy
-mm-dd'),'1900-01-01') then 1 else 0 endif) as F from vacancy order by
whenentered desc;
expect departmentid=~~,employmentid=TEST,D=1,status=],position=test,E=1,F=1;
insert into iqxnetrightallowed values ('AUTHORISEJOBS','CLIENT','*',1);
select * from
NetClientAddVacancies('test.client','test',dateformat(now(*),'dd/mm/yyyy'),d
ateformat(dateadd(dd,3,now(*)),'dd/mm/yyyy'),null,null,null,1);
expect pResult=0*;
update vacancy set status=null where vacancyid='test';

select top 1 staffid,employmentid,replace(description,' ','!') as
A,DurationType from diary order by whenentered desc ;
expect staffid=TEST,A=New!vacancy*,DurationType=R;
update vacancy set status=[' where vacancyid='test';
*/
GO
```

```
COMMENT TO PRESERVE FORMAT ON PROCEDURE "pears"."NetClientAddVacancies" IS
{CREATE PROCEDURE pears."NetClientAddVacancies"( IN "pWebUserID" CHAR(20),IN
"ptemplatevacancyid" CHAR(50),IN "pdatefrom" CHAR(25),
IN "pdateto" CHAR(25) DEFAULT NULL,IN "pTheirRef" CHAR(50) DEFAULT NULL,IN
"pTitle" CHAR(100) DEFAULT NULL,IN "pDescription" long VARCHAR DEFAULT
NULL,IN "pquantity" CHAR(25) DEFAULT '1' )
RESULT( "pResult" CHAR(250) )
// IQXWeb
BEGIN
DECLARE "swebusername" CHAR(50);
DECLARE "sid" CHAR(20);
DECLARE "cnt" INTEGER;
DECLARE "dfrom" DATE;
DECLARE "dto" DATE;
DECLARE "CanAuthorise" SMALLINT;
DECLARE "NewVacancyState" CHAR(1);
DECLARE LOCAL TEMPORARY TABLE "IDs"(
"ID" CHAR(20) NULL,
) NOT transactional;
INSERT INTO "IDs"( "ID" )
SELECT "companyid" FROM "employment" KEY JOIN "iqxnetuserlink" WHERE
"iqxnetuserid" = "pWebUserID";
IF NOT "ptemplatevacancyid" = any(SELECT "v"."vacancyid" FROM "vacancy" AS
"v" KEY JOIN "employment" AS "e" JOIN "IDs" ON "e"."companyid" = "IDs"."ID")
THEN
SELECT '99:~Permission denied';
RETURN
```



```
END IF;
IF "IQXNetHasPermission"("pWebUserID", 'ORDERJOBS') = 0 THEN
    SELECT '99:~Permission denied';
    RETURN
END IF;
SET "swebusername" = (SELECT "name" FROM "iqxnetuser" WHERE "iqxnetuserid"
= "pwebuserid");
SET "cnt" = "iqxnetstringtointeger"("pquantity");
IF "cnt" IS NULL OR "cnt" = 0 THEN
    SELECT '450:~Invalid quantity';
    RETURN
END IF;
SET "dfrom" = "iqxnetstringtodate"("pdatefrom");
IF "dfrom" IS NULL THEN
    SELECT '500:~Invalid start date';
    RETURN
END IF;
SET "pdateto" = "nullif"("trim"("pdateto"), '');
SET "dto" = "iqxnetstringtodate"("pdateto");
IF "pdateto" IS NOT NULL AND "dto" IS NULL THEN
    SELECT '500:~Invalid end date';
    RETURN
END IF;
SET "CanAuthorise" = "IQXNetHasPermission"("pWebUserID", 'AUTHORISEJOBS');
IF "CanAuthorise" = 1 THEN
    SET "NewVacancyState" = 'C'
ELSE
    SET "NewVacancyState" = ']'
END IF;
SET "sid" = "uniquekey"("cnt");
INSERT INTO "vacancy"(
"vacancyid", "departmentid", "employmentid", "staffid", "entrydate", "status", "st
artdate", "FinishDate", "temp", "position", "notes", "othernotes", "classcode", "Er
NI", "HolidayAllowance", "Discount",
"TempDeskID", "PayrollIdentifier", "AnalysisCode", "clientdepartment", "Currency
", "RecipientID", "Role", "SiteName", "SiteContact", "addr1", "addr2", "addr3", "tow
n", "county", "country", "postcode", "TempJobTypeID",
"SitePhoneNumbers", "Invaddr1", "Invaddr2", "Invaddr3", "Invtown", "Invcounty", "I
nvcountry", "Invpostcode", "SiteFax", "SiteEmail", "TheirRef",
"PriceListID", "SendShiftsToPayroll", "WorkMonday", "WorkTuesday", "WorkWednesda
y", "WorkThursday", "WorkFriday", "WorkSaturday", "WorkSunday", "WorkNormalHours"
, "WorkStartTime", "noofposts" )
    SELECT "sid", "departmentid", "employmentid", "staffid", CURRENT
DATE, "NewVacancyState", "dfrom", "dto", "temp", "isnull"("pTitle", "position"), "n
otes", "isnull"("pDescription", "othernotes"), "classcode", "ErNI", "HolidayAllow
ance", "Discount",
"TempDeskID", "PayrollIdentifier", "AnalysisCode", "clientdepartment", "Currency
", "RecipientID", "Role", "SiteName", "SiteContact", "addr1", "addr2", "addr3", "tow
```



```
n","county","country","postcode","TempJobTypeID",
"SitePhoneNumbers","Invaddr1","Invaddr2","Invaddr3","Invtown","Invcounty","I
nvcountry","Invpostcode","SiteFax","SiteEmail","ucase"("pTheirRef"),
"PriceListID","SendShiftsToPayroll","WorkMonday","WorkTuesday","WorkWednesda
y","WorkThursday","WorkFriday","WorkSaturday","WorkSunday","WorkNormalHours"
,"WorkStartTime",cnt
    FROM "vacancy" WHERE "vacancyid" = "ptemplatevacancyid";
INSERT INTO "tempjobrate"(
"TempJobRateID","VacancyID","TempPayBandID","PayRate","ChargeRate","StartDat
e","EndDate","Grade" )
SELECT
"uniquekey"("TempJobRateID"),"sid","TempPayBandID","PayRate","ChargeRate","S
tartDate","EndDate","Grade"
    FROM "tempjobrate" WHERE "vacancyid" = "ptemplatevacancyid";
INSERT INTO "criterion"(
"searchlocation","id","critid","sourcelocation","dictionaryid","tagid","tagc
hoiceid","notflag","matchtype","textvalue","value","uppervalue","taganytotal
","extravalue","extrauppervalue","extramatchtype" )
SELECT
"searchlocation","sid","critid","sourcelocation","dictionaryid","tagid","tag
choiceid","notflag","matchtype","textvalue","value","uppervalue","taganytota
l","extravalue","extrauppervalue","extramatchtype"
    FROM "criterion" WHERE "searchlocation" LIKE 'V%' AND "id" =
"ptemplatevacancyid";
CALL "NetSetVacancyCascade"("sid");
IF "CanAuthorise" = 1 THEN
CALL
"IQXNetPopup"("pWebUserID",NULL,NULL,"sid",NULL,NULL,'ADDVACANCY','New
vacancy requirement added',NULL)
END IF;
SELECT '0:~Success'
END
```

/* DOC

2016-10-25 PC IW-69 test and doc sections added

2018-08-22 PC improve test handling

2019-02-20 PC IW-838 create only one vacancy with multiple count, instead of
looping to create multiple vacancies

tests:

invalid pWebUserID

valid pWebUserID and invalid pTemplateVacancyID

valid pWebUserID, valid pTemplateVacancyID which does not belong to
pWebUserID

valid pWebUserID, valid pTemplateVacancyID but ORDERJOBS set false

valid pWebUserID, valid pTemplateVacancyID, ORDERJOBS set true but

invalid quantity

valid pWebUserID, valid pTemplateVacancyID, ORDERJOBS set true, valid



```
quantity but invalid start date
    valid pWebUserID, valid pTemplateVacancyID, ORDERJOBS set true, valid
quantity, valid start date but invalid end date
    valid pWebUserID, valid pTemplateVacancyID, ORDERJOBS set true, valid
quantity, valid start date and null end date
    valid pWebUserID, valid pTemplateVacancyID, ORDERJOBS set true, valid
quantity, valid start date and null valid date
    all fields valid and AUTHORISEJOBS set to 0
    all fields valid and AUTHORISEJOBS set to 1

*/

/* 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.client';
update vacancy set status='[' where vacancyid='test';
delete from iqxnetrightallowed where iqxnetrightid='ORDERJOBS' and
iqxnetuserclassid='client';
delete from iqxnetrightallowed where iqxnetrightid='AUTHORISEJOBS' and
iqxnetuserclassid='client';
select * from
NetClientAddVacancies(null,null,null,null,null,null,null,null);
expect pResult=99*;
select * from
NetClientAddVacancies('test.x',null,null,null,null,null,null,null);
expect pResult=99*;
select * from
NetClientAddVacancies('test.client',null,null,null,null,null,null,null);
expect pResult=99*;
insert into iqxnetrightallowed values ('ORDERJOBS','CLIENT','*',1);
select * from
NetClientAddVacancies('test.client','test',null,null,null,null,null,null);
expect pResult=450*;
select * from
NetClientAddVacancies('test.client','test',null,null,null,null,null,1);
expect pResult=500*;
select * from
NetClientAddVacancies('test.client','test',dateformat(now(*),'dd/mm/yyyy'),'
xx',null,null,null,1);
expect pResult=500*;
select * from
NetClientAddVacancies('test.client','test',dateformat(now(*),'dd/mm/yyyy'),n
ull,null,null,null,1);
expect pResult=0*;
select top 1 departmentid,employmentid,(if
```



```
entrydate=dateformat(now(*),'yyyy-mm-dd') then 1 else 0 endif) as
A,status,position,(if startdate=dateformat(now(*),'yyyy-mm-dd') then 1 else
0 endif) as B,(if
isnull(finishdate,'1900-01-01')=isnull(dateformat(dateadd(dd,2,now(*)),'yyyy
-mm-dd'),'1900-01-01') then 1 else 0 endif) as C from vacancy order by
whenentered desc;
expect departmentid=~~,employmentid=TEST,A=1,status=],position=test,B=1,C=0;
select * from
NetClientAddVacancies('test.client','test',dateformat(now(*),'dd/mm/yyyy'),d
ateformat(dateadd(dd,2,now(*)),'dd/mm/yyyy'),null,null,null,1);
expect pResult=0*;
select top 1 departmentid,employmentid,(if
entrydate=dateformat(now(*),'yyyy-mm-dd') then 1 else 0 endif) as
D,status,position,(if startdate=dateformat(now(*),'yyyy-mm-dd') then 1 else
0 endif) as E,(if
isnull(finishdate,'1900-01-01')=isnull(dateformat(dateadd(dd,2,now(*)),'yyyy
-mm-dd'),'1900-01-01') then 1 else 0 endif) as F from vacancy order by
whenentered desc;
expect departmentid=~~,employmentid=TEST,D=1,status=],position=test,E=1,F=1;
insert into iqxnetrightallowed values ('AUTHORISEJOBS','CLIENT','*',1);
select * from
NetClientAddVacancies('test.client','test',dateformat(now(*),'dd/mm/yyyy'),d
ateformat(dateadd(dd,3,now(*)),'dd/mm/yyyy'),null,null,null,1);
expect pResult=0*;
update vacancy set status=null where vacancyid='test';

select top 1 staffid,employmentid,replace(description,' ','!') as
A,DurationType from diary order by whenentered desc ;
expect staffid=TEST,A=New!vacancy*,DurationType=R;
update vacancy set status='[' where vacancyid='test';
*/
}
```

From:

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

Permanent link:

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

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

