



pears.WithHolds



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

Description

Stores withholds. When a worker is blocked from a vacancy or a company

Columns

Column	Type	Null	Default	Comment
WithHoldsID	char(20)	NOT NULL		
PersonID	char(20)	NULL		
CompanyID	char(20)	NULL		
VacancyID	char(20)	NULL		
Note	long varchar	NULL		
StaffID	char(20)	NULL		
ReasonID	char(20)	NULL		
WhenCreated	timestamp	NULL	current timestamp	

Primary Key

- WithHoldsID

Foreign Keys

Constraint	Columns	References	Delete/update action
person	PersonID	pears.Person (personid)	ON DELETE CASCADE
company	CompanyID	pears.Company (companyid)	ON DELETE CASCADE
vacancy	VacancyID	pears.vacancy (vacancyid)	ON DELETE CASCADE
staff	StaffID	pears.staff (staffid)	ON DELETE CASCADE
WithHoldReason	ReasonID	pears.WithHoldReason (ReasonID)	ON DELETE SET NULL

Referenced By

- No incoming foreign keys found.



Indexes

- No indexes found.

Triggers

Name	Timing	Event
Withholds_UpdateAudit	before	update of "reasonid" order 1
Withholds_DeleteAudit	before	delete order 1

Original SQL

```
-- IQX database structure split by table
-- Source: IQXDatabaseStructure - with comments.sql
-- Table: "pears"."WithHolds"
-- Table comment: Stores withholds. When a worker is blocked from a vacancy
or a company
-- Statement count: 11

CREATE TABLE "pears"."WithHolds" (
    "WithHoldsID"          CHAR(20) NOT NULL
    , "PersonID"            CHAR(20) NULL
    , "CompanyID"           CHAR(20) NULL
    , "VacancyID"           CHAR(20) NULL
    , "Note"                long VARCHAR NULL
    , "StaffID"             CHAR(20) NULL
    , "ReasonID"            CHAR(20) NULL
    , "WhenCreated"         TIMESTAMP NULL DEFAULT CURRENT
TIMESTAMP
    , PRIMARY KEY ("WithHoldsID" ASC)
)
GO

COMMENT ON TABLE "pears"."WithHolds" IS
    'Stores withholds. When a worker is blocked from a vacancy or a company'
GO

ALTER TABLE "pears"."WithHolds"
    ADD FOREIGN KEY "person" ("PersonID" ASC)
    REFERENCES "pears"."Person" ("personid")
    ON DELETE CASCADE
GO
```



```
ALTER TABLE "pears"."WithHolds"
  ADD FOREIGN KEY "company" ("CompanyID" ASC)
  REFERENCES "pears"."Company" ("companyid")
  ON DELETE CASCADE
GO

ALTER TABLE "pears"."WithHolds"
  ADD FOREIGN KEY "vacancy" ("VacancyID" ASC)
  REFERENCES "pears"."vacancy" ("vacancyid")
  ON DELETE CASCADE
GO

ALTER TABLE "pears"."WithHolds"
  ADD FOREIGN KEY "staff" ("StaffID" ASC)
  REFERENCES "pears"."staff" ("staffid")
  ON DELETE CASCADE
GO

ALTER TABLE "pears"."WithHolds"
  ADD FOREIGN KEY "WithHoldReason" ("ReasonID" ASC)
  REFERENCES "pears"."WithHoldReason" ("ReasonID")
  ON DELETE SET NULL
GO

CREATE TRIGGER "Withholds_UpdateAudit" BEFORE UPDATE OF "reasonid"
ORDER 1 ON
"pears"."Withholds"
REFERENCING OLD AS "oldw" NEW AS "neww"
FOR each ROW
BEGIN
  DECLARE "NewDesc" CHAR(200);
  DECLARE "OldDesc" CHAR(200);
  DECLARE "PName" CHAR(200);
  SELECT "string"(CASE "type" WHEN 'T' THEN 'Candidate: ' WHEN 'C' THEN
'Client: ' WHEN 'A' THEN 'Agency: ' END,"description")
  INTO "OldDesc" FROM "WithholdReason" WHERE "reasonid" =
"oldw"."reasonid";
  SELECT "string"(CASE "type" WHEN 'T' THEN 'Candidate: ' WHEN 'C' THEN
'Client: ' WHEN 'A' THEN 'Agency: ' END,"description")
  INTO "NewDesc" FROM "WithholdReason" WHERE "reasonid" =
"neww"."reasonid";
  SET "PName" = "string"((SELECT "name" FROM "person" WHERE "personid" =
"neww"."personid"),' ',(SELECT "name" FROM "company" WHERE "companyid" =
```



```
"neww"."companyid"),
    ' ',(SELECT "position"+' '+'refcode" FROM "vacancy" WHERE "vacancyid" =
"neww"."vacancyid"));
    CALL "AuditLog"('PERSON',"neww"."PersonID",'Withhold Reason edited
'+PName,"OldDesc","NewDesc");
    CALL "AuditLog"('COMPANY',"neww"."CompanyID",'Withhold Reason edited
'+PName,"OldDesc","NewDesc");
    IF "neww"."VacancyID" IS NOT NULL THEN CALL
"AuditLog"('VACANCY',"neww"."VacancyID",'Withhold Reason edited
'+PName,"OldDesc","NewDesc")
    END IF
END
GO

COMMENT TO PRESERVE FORMAT ON TRIGGER
"pears"."WithHolds"."Withholds_UpdateAudit" IS
{CREATE TRIGGER Withholds_UpdateAudit
    BEFORE UPDATE OF reasonid ORDER 1 ON
pears.Withholds
REFERENCING NEW AS neww OLD AS oldw
FOR each ROW
BEGIN
    DECLARE NewDesc CHAR(200);
    DECLARE OldDesc CHAR(200);
    DECLARE PName CHAR(200);
    SELECT string( CASE TYPE WHEN 'T' THEN 'Candidate: ' WHEN 'C' THEN
'Client: ' WHEN 'A' THEN 'Agency: ' END, description)
        INTO OldDesc FROM WithholdReason WHERE reasonid = oldw.reasonid;
    SELECT string( CASE TYPE WHEN 'T' THEN 'Candidate: ' WHEN 'C' THEN
'Client: ' WHEN 'A' THEN 'Agency: ' END, description)
        INTO NewDesc FROM WithholdReason WHERE reasonid = neww.reasonid;
    SET PName = string((SELECT name FROM person WHERE personid =
neww.personid),' ',(SELECT name FROM company WHERE companyid =
neww.companyid),
    ' ',(SELECT POSITION+' '+'refcode" FROM vacancy WHERE vacancyid =
neww.vacancyid));
    CALL AuditLog('PERSON',neww.PersonID,'Withhold Reason edited
'+PName,OldDesc,NewDesc);
    CALL AuditLog('COMPANY',neww.CompanyID,'Withhold Reason edited
'+PName,OldDesc,NewDesc);
    IF neww.VacancyID IS NOT NULL THEN CALL
AuditLog('VACANCY',neww.VacancyID,'Withhold Reason edited
'+PName,OldDesc,NewDesc) END IF
END
}
GO
```



```
CREATE TRIGGER "Withholds_DeleteAudit" BEFORE DELETE ORDER 1 ON
"pears"."Withholds"
REFERENCING OLD AS "oldw"
FOR each ROW
BEGIN
    DECLARE "OldDesc" CHAR(200);
    DECLARE "PName" CHAR(200);
    SELECT "string"(CASE "type" WHEN 'T' THEN 'Candidate: ' WHEN 'C' THEN
'Client: ' WHEN 'A' THEN 'Agency: ' END,"description")
        INTO "OldDesc" FROM "WithholdReason" WHERE "reasonid" =
"oldw"."reasonid";
    SET "PName" = "string"((SELECT "name" FROM "person" WHERE "personid" =
"oldw"."personid"),' ',(SELECT "name" FROM "company" WHERE "companyid" =
"oldw"."companyid"),
        ' ',(SELECT "position"+" "+refcode FROM "vacancy" WHERE "vacancyid" =
"oldw"."vacancyid"));
    CALL "AuditLog"('PERSON',"oldw"."PersonID",'Withhold Reason deleted
'+PName,"OldDesc",' ');
    CALL "AuditLog"('COMPANY',"oldw"."CompanyID",'Withhold Reason deleted
'+PName,"OldDesc",' ');
    IF "oldw"."VacancyID" IS NOT NULL THEN CALL
"AuditLog"('VACANCY',"oldw"."VacancyID",'Withhold Reason deleted
'+PName,"OldDesc",' ') END IF
END
GO
```

```
COMMENT TO PRESERVE FORMAT ON TRIGGER
"pears"."WithHolds"."Withholds_DeleteAudit" IS
{CREATE TRIGGER Withholds_DeleteAudit
    BEFORE DELETE ORDER 1 ON
pears.Withholds
REFERENCING OLD AS oldw
FOR each ROW
BEGIN
    DECLARE OldDesc CHAR(200);
    DECLARE PName CHAR(200);
    SELECT string( CASE TYPE WHEN 'T' THEN 'Candidate: ' WHEN 'C' THEN
'Client: ' WHEN 'A' THEN 'Agency: ' END, description)
        INTO OldDesc FROM WithholdReason WHERE reasonid = oldw.reasonid;
    SET PName = string((SELECT name FROM person WHERE personid =
oldw.personid),' ',(SELECT name FROM company WHERE companyid =
oldw.companyid),
        ' ',(SELECT POSITION+' '+refcode FROM vacancy WHERE vacancyid =
oldw.vacancyid));
    CALL AuditLog('PERSON',oldw.PersonID,'Withhold Reason deleted
'+PName,OldDesc,' ');
```



```
CALL AuditLog('COMPANY',oldw.CompanyID,'Withhold Reason deleted
'+PName,OldDesc,' ');
IF oldw.VacancyID IS NOT NULL THEN CALL
AuditLog('VACANCY',oldw.VacancyID,'Withhold Reason deleted '+PName,OldDesc,'
') END IF;
END
}
GO
```

From:

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

Permanent link:

https://iqxusers.co.uk/iqxhelp/doku.php?id=database:tables:pears_withholds

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

