



pears.TempPoolMember



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

Description

Link between candidates and TempDesks.

Columns

Column	Type	Null	Default	Comment
PersonID	char(20)	NOT NULL		
TempDeskID	char(20)	NOT NULL		

Primary Key

- PersonID, TempDeskID

Foreign Keys

Constraint	Columns	References	Delete/update action
Person	PersonID	pears.Person (personid)	NOT NULL; ON DELETE CASCADE
TempDesk	TempDeskID	pears.tempdesk (tempdeskid)	NOT NULL; ON DELETE CASCADE

Referenced By

- No incoming foreign keys found.

Indexes

- No indexes found.



Triggers

Name	Timing	Event
TempPoolMemberDeleteAudit	before	delete order 2
TempPoolMemberAddAudit	after	insert order 1

Original SQL

```
-- IQX database structure split by table
-- Source: IQXDatabaseStructure - with comments.sql
-- Table: "pears"."TempPoolMember"
-- Table comment: Link between candidates and TempDesks.
-- Statement count: 8

CREATE TABLE "pears"."TempPoolMember" (
    "PersonID" CHAR(20) NOT NULL
    , "TempDeskID" CHAR(20) NOT NULL
    , PRIMARY KEY ("PersonID" ASC, "TempDeskID" ASC)
)
GO

COMMENT ON TABLE "pears"."TempPoolMember" IS
    'Link between candidates and TempDesks.'
GO

ALTER TABLE "pears"."TempPoolMember"
    ADD NOT NULL FOREIGN KEY "Person" ("PersonID" ASC)
    REFERENCES "pears"."Person" ("personid")
    ON DELETE CASCADE
GO

ALTER TABLE "pears"."TempPoolMember"
    ADD NOT NULL FOREIGN KEY "TempDesk" ("TempDeskID" ASC)
    REFERENCES "pears"."tempdesk" ("tempdeskid")
    ON DELETE CASCADE
GO

CREATE TRIGGER "TempPoolMemberDeleteAudit" BEFORE DELETE ORDER 2 ON
    "pears"."TempPoolMember"
    REFERENCING OLD AS "old_name"
    FOR each ROW
    BEGIN
```



```
DECLARE "DeskName" CHAR(50);
SELECT "name" INTO "DeskName" FROM "TempDesk" WHERE "tempdeskid" =
"old_name"."tempdeskid";
CALL "AuditLog"('PERSON',"old_name"."personid","string"('Desk
',"DeskName",' Removed '), 'Removed', '')
END
GO

COMMENT TO PRESERVE FORMAT ON TRIGGER
"pears"."TempPoolMember"."TempPoolMemberDeleteAudit" IS
{CREATE TRIGGER TempPoolMemberDeleteAudit
BEFORE DELETE ORDER 2 ON
pears.TempPoolMember
REFERENCING OLD AS old_name
FOR each ROW
BEGIN
DECLARE DeskName CHAR(50);
SELECT name INTO DeskName FROM TempDesk WHERE tempdeskid =
old_name.tempdeskid;
CALL AuditLog('PERSON',old_name.personid,string('Desk ',DeskName,' Removed
'),'Removed', '')
END
}
GO

CREATE TRIGGER "TempPoolMemberAddAudit" after INSERT ORDER 1 ON
"pears"."TempPoolMember"
REFERENCING NEW AS "new_name"
FOR each ROW
BEGIN
DECLARE "DeskName" CHAR(50);
SELECT "name" INTO "DeskName" FROM "TempDesk" WHERE "tempdeskid" =
"new_name"."tempdeskid";
CALL "AuditLog"('PERSON',"new_name"."personid","string"('Desk
',"DeskName",' Added '), 'Added', '')
END
GO

COMMENT TO PRESERVE FORMAT ON TRIGGER
"pears"."TempPoolMember"."TempPoolMemberAddAudit" IS
{CREATE TRIGGER TempPoolMemberAddAudit
after INSERT ORDER 1 ON
pears.TempPoolMember
REFERENCING NEW AS new_name
FOR each ROW
```



```
BEGIN
  DECLARE DeskName CHAR(50);
  SELECT name INTO DeskName FROM TempDesk WHERE tempdeskid =
new_name.tempdeskid;
  CALL AuditLog('PERSON',new_name.personid,string('Desk ',DeskName,' Added
'),'Added','')
END
}
GO
```

From:

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

Permanent link:

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

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

