



pears.PersonAvailableforPlan



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

Original SQL

```
CREATE FUNCTION "pears"."PersonAvailableforPlan"(  
  /* Application Maintained Function / Procedure - DO NOT EDIT*/  
  IN "sPersonid" CHAR(20),IN "dShiftDate" DATE,IN "tShiftFrom" TIME,IN  
  "tShiftTo" TIME,IN "iRecovery" SMALLINT,IN "iMinutes" SMALLINT,  
  IN "iCheckExplicitlyAvailable" SMALLINT,IN "departmentid" CHAR(2),IN  
  "EssentialSkill" CHAR(15),IN "EssentialSkillGradeID" CHAR(10),IN  
  "EssentialSkillChoiceList" CHAR(100),IN "MovingShiftID" CHAR(20) )  
RETURNS CHAR(20)  
BEGIN  
  DECLARE "foundState" CHAR(20);  
  DECLARE "founddigit" CHAR(1);  
  DECLARE "slotfound" tinyint;  
  DECLARE "MustBeKnownAvailable" tinyint;  
  DECLARE "RequiredGradeValue" INTEGER;  
  DECLARE "UpperGradeValue" INTEGER;  
  DECLARE "bExactGrade" INTEGER;  
  DECLARE "MaxGradeValue" INTEGER;  
  DECLARE "checktime" TIME;  
  DECLARE "tloc" CHAR(3);  
  DECLARE "tid" CHAR(3);  
  DECLARE "gtid" CHAR(3);  
  DECLARE "tcid" CHAR(4);  
  DECLARE "i" INTEGER;  
  SET "foundstate" = NULL;  
  IF "spersonid" IS NULL THEN  
    RETURN NULL  
  END IF;  
  SET "maxgradevalue" = 1000000000; -- 1 billion should be safe  
  IF "left"("essentialskillgradeid",2) = '==' THEN  
    SET "bExactGrade" = 1;  
    SET "essentialskillgradeid" = "substring"("essentialskillgradeid",3)  
  ELSE  
    SET "bExactGrade" = 0  
  END IF;  
  IF "icheckexplicitlyavailable" < 0 THEN
```



```
SET "mustbeknownavailable" = 0;
IF "icheckexplicitlyavailable" = -1 THEN
    SET "icheckexplicitlyavailable" = 1
ELSE
    SET "icheckexplicitlyavailable" = 0
END IF
ELSE SELECT "onlymatchifknownavailable" INTO "mustbeknownavailable" FROM
"person" WHERE "personid" = "spersonid";
SET "mustbeknownavailable" = "isnull"("mustbeknownavailable",0);
IF "mustbeknownavailable" = 1 THEN
    SET "icheckexplicitlyavailable" = 1
END IF END IF;
IF "length"("isnull"("essentialskill","')) > 2 THEN
SET "i" = "charindex(';', "essentialskill");
IF "i" > 0 THEN
    SET "tloc" = "left"("essentialskill", "i"-1);
    SET "essentialskill" =
"right"("essentialskill", "length"("essentialskill")-"i")
END IF;
SET "i" = "charindex(';', "essentialskill");
IF "i" > 0 THEN
    SET "tid" = "left"("essentialskill", "i"-1);
    SET "essentialskill" =
"right"("essentialskill", "length"("essentialskill")-"i")
END IF;
SET "i" = "charindex(';', "essentialskill");
IF "i" > 0 THEN
    SET "tcid" = "left"("essentialskill", "i"-1);
    SET "gtid" = "right"("essentialskill", "length"("essentialskill")-"i");
    IF "trim"("gtid") = '' THEN
        SET "gtid" = "tid"
    END IF
ELSE SET "tcid" = "essentialskill";
    SET "gtid" = "tid"
END IF;
IF "tloc" = 'A' THEN
    SET "tloc" = "string"('A', "departmentid")
END IF;
IF "gtid" <> "tid" THEN // gtid contains a single SELECT grade question.
The choice VALUE (OR sortorder IF VALUE IS NULL) IS used TO rate the grade
    SELECT "isnull"("value", "sortorder") INTO "requiredgradevalue" FROM
"tagchoice" WHERE "taglocation" = "tloc" AND "tagid" = "gtid" AND
"tagchoiceid" = "essentialskillgradeid" AND "isnull"("subchoice",0) = 0;
    IF "requiredgradevalue" IS NULL THEN
        SET "requiredgradevalue" = 1;
        SET "uppergradevalue" = "maxgradevalue"
    ELSE
        IF "bExactGrade" = 1 THEN
```



```
        SET "uppergradevalue" = "requiredgradevalue"
    ELSE
        SET "uppergradevalue" = "maxgradevalue"
    END IF END IF;
    IF NOT EXISTS(SELECT * FROM "tagvalue" AS "v" KEY JOIN "tagchoice" AS
"c" WHERE "v"."taglocation" = "tloc" AND "v"."tagid" = "gtid" AND "v"."id" =
"spersonid" AND "isnull"("c"."value","c"."sortorder") BETWEEN
"requiredgradevalue" AND "uppergradevalue") THEN
        RETURN('Q') // NOT qualified
    END IF;
    SET "requiredgradevalue" = 1; // Grade CHECK done. The skill selection
may be multiple choice - any positive VALUE will do
    SET "uppergradevalue" = "maxgradevalue"
    ELSE
        SELECT "value" INTO "requiredgradevalue" FROM "tagchoice" WHERE
"taglocation" = "tloc" AND "tagid" = "tid" AND "tagchoiceid" =
"essentialskillgradeid" AND "subchoice" = 1;
        IF "requiredgradevalue" IS NULL THEN
            SET "requiredgradevalue" = 1;
            SET "uppergradevalue" = "maxgradevalue"
        ELSE
            IF "bExactGrade" = 1 THEN
                SET "uppergradevalue" = "requiredgradevalue"
            ELSE
                SET "uppergradevalue" = "maxgradevalue"
            END IF
        END IF END IF;
    IF "tcid" <> '' THEN
        SET "essentialskillchoicelist" = "tcid"
    ELSE
        SET "essentialskillchoicelist" =
"isnull"("essentialskillchoicelist",'')
    END IF;
    while "essentialskillchoicelist" <> '' loop
        SET "i" = "charindex"(';','essentialskillchoicelist');
        IF "i" = 0 THEN
            SET "tcid" = "essentialskillchoicelist";
            SET "essentialskillchoicelist" = ''
        ELSE
            SET "tcid" = "left"("essentialskillchoicelist","i"-1);
            SET "essentialskillchoicelist" =
"right"("essentialskillchoicelist","length"("essentialskillchoicelist")-"i")
        END IF;
        IF NOT EXISTS(SELECT * FROM "tagvalue" WHERE "taglocation" = "tloc"
AND "tagid" = "tid" AND "tagchoiceid" LIKE "tcid"
AND "id" = "spersonid" AND "value" BETWEEN "requiredgradevalue"
AND "uppergradevalue") THEN
            RETURN('Q') // NOT qualified
```



```
END IF
END loop END IF;
SET "movingshiftid" = "isnull"("movingshiftid",'');
// FIRST CHECK FOR ALL DAY unavailability
SELECT FIRST "state" INTO "foundstate" FROM "tempshift" WHERE "state" NOT
IN( 'A','C' ) AND "personid" = "spersonid" AND "tempshiftid" <>
"movingshiftid" AND "shiftdate" = "dshiftdate" AND("timefrom" IS NULL OR
"timeeto" IS NULL);
IF "foundstate" IS NULL THEN // Look FOR non-concurrent employments
SELECT FIRST 'W' INTO "foundstate" FROM "employment" WHERE
"isnull"("concurrent",0) = 0 AND "personid" = "spersonid" AND "startdate" <=
"dshiftdate" AND("leavedate" IS NULL OR "leavedate" >= "dshiftdate")
END IF;
IF "foundstate" IS NULL THEN
IF "isnull"("iminutes",0) = 0 OR "iminutes" > 480 OR "tshiftto" <=
"tshiftfrom" THEN // Do NOT do iterative MATCH FOR overnight shifts
SET "foundstate" =
"personavailableforplaniterate"("spersonid","dshiftdate","tshiftfrom","tshif
tto","icheckexplicitlyavailable","movingshiftid");
SET "foundstate" =
"personavailableforrecovery"("foundstate","sPersonid","dShiftDate","tShiftFr
om","tShiftTo","iRecovery","MovingShiftID")
ELSE
SET "slotfound" = 0;
SET "checktime" = "tshiftto";
SET "tshiftto" = CAST("dateadd"("minute","iminutes","tshiftfrom") AS
TIME);
"iterator": loop
SET "founddigit" =
"personavailableforplaniterate"("spersonid","dshiftdate","tshiftfrom","tshif
tto","icheckexplicitlyavailable","movingshiftid");
SET "founddigit" =
"personavailableforrecovery"("founddigit","sPersonid","dShiftDate","tShiftFr
om","tShiftTo","iRecovery","MovingShiftID");
IF "founddigit" IS NULL AND "slotfound" = 0 THEN
SET "foundstate" =
"string"('M',"dateformat"("dshiftdate",'yyyymmdd'),'dateformat"("tshiftfrom"
,'hhnn'),'dateformat"("tshiftto",'hhnn')));
SET "slotfound" = 1;
IF "icheckexplicitlyavailable" = 0 THEN
leave "iterator"
END IF
ELSE IF "founddigit" = 'A' THEN
SET "foundstate" =
"string"('A',"dateformat"("dshiftdate",'yyyymmdd'),'dateformat"("tshiftfrom"
,'hhnn'),'dateformat"("tshiftto",'hhnn')));
leave "iterator"
ELSE IF "foundstate" IS NULL AND "founddigit" IN(
```



```
'H','U','P','B','W' ) THEN
    SET "foundstate" = "founddigit"
    END IF END IF END IF;
SET "tshiftfrom" = CAST("dateadd"("minute",30,"tshiftfrom") AS
TIME);
SET "tshiftto" = CAST("dateadd"("minute",30,"tshiftto") AS TIME);
IF "tshiftto" > "checktime" OR "tshiftto" <= "tshiftfrom" THEN
    leave "iterator"
END IF
END loop "iterator" END IF END IF;
IF("foundstate" IS NULL OR "foundstate" LIKE 'M%') AND
"mustbeknownavailable" = 1 THEN
    SET "foundstate" = 'U'
END IF;
RETURN "foundstate"
END
```

From:

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

Permanent link:

https://iqxusers.co.uk/iqxhelp/doku.php?id=database:functions:pears_personavailableforplan

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

