# **Validation on HR Special Info (SIT) in SSHR Pages**

**Validation on HR Special Info (SIT) in SSHR Pages:**

User hooks doesn’t work on SIT SSHR Pages for validations but works once the final approval is done.

This is known bug.

To overcome above issue we will implement logic in FND\_FLEX\_PLSQL package.

Requirement : To make validation on SIT SSHR page so the user can update once and submit for approval, once the user update the same record twice which is under approval, then the error message should appear.

New update.........  
  
     0. Create one new segment in the structure named 'sequence' not mandatory and hidden.  
    valuset10 char, default type so,

SELECT F\_FND\_FLEX\_S.NEXTVAL FROM DUAL WHERE : $FLEX$.FND\_DATE4\_REQUIRED2 IS NOT NULL  
  
f\_fnd\_flex\_s is the sequence in the database created custom.  
FND\_DATE4\_REQUIRED2 is the value set for the resumption date, once this date changed/updated the sequence will update auto.  
  
as the package FND\_FLEX\_PLSQL mentioned in the step7 will validate/fires only when there is no code combination exists, so we need to have difference code combinations by generating the sequence to validate every time.  
  
Oracle reference: How to avoid that FND\_FLEX\_PLSQL.VALIDATE () is not triggered when the same data is re-entered? (Doc ID 2023950.1)

CREATE OR REPLACE PACKAGE BODY APPS.FND\_FLEX\_PLSQL AS

*/\* $Header: AFFFPLVB.pls 115.0 99/07/16 23:18:46 porting ship $ \*/*

FUNCTION validate(application\_id IN NUMBER,

id\_flex\_code IN VARCHAR2,

id\_flex\_num IN NUMBER,

vdate IN DATE,

segment\_delimiter IN VARCHAR2,

concatenated\_segments IN VARCHAR2,

numsegs IN NUMBER,

user\_segment1 IN VARCHAR2,

user\_segment2 IN VARCHAR2,

user\_segment3 IN VARCHAR2,

user\_segment4 IN VARCHAR2,

user\_segment5 IN VARCHAR2,

user\_segment6 IN VARCHAR2,

user\_segment7 IN VARCHAR2,

user\_segment8 IN VARCHAR2,

user\_segment9 IN VARCHAR2,

user\_segment10 IN VARCHAR2,

user\_segment11 IN VARCHAR2,

user\_segment12 IN VARCHAR2,

user\_segment13 IN VARCHAR2,

user\_segment14 IN VARCHAR2,

user\_segment15 IN VARCHAR2,

user\_segment16 IN VARCHAR2,

user\_segment17 IN VARCHAR2,

user\_segment18 IN VARCHAR2,

user\_segment19 IN VARCHAR2,

user\_segment20 IN VARCHAR2,

user\_segment21 IN VARCHAR2,

user\_segment22 IN VARCHAR2,

user\_segment23 IN VARCHAR2,

user\_segment24 IN VARCHAR2,

user\_segment25 IN VARCHAR2,

user\_segment26 IN VARCHAR2,

user\_segment27 IN VARCHAR2,

user\_segment28 IN VARCHAR2,

user\_segment29 IN VARCHAR2,

user\_segment30 IN VARCHAR2,

error\_message OUT VARCHAR2)

return BOOLEAN

IS

FL\_PERSONID VARCHAR2(200):= FND\_PROFILE.VALUE('PER\_PERSON\_ID');

FL\_SHARE\_PERCENTAGE NUMBER;

BEGIN

*/\**

*VALIDATION FOR SIT OF EMPLOYEES*

*FUND NOMEENI SHARE PERCENTAGE CANNT GREATER THAN 100 AND LESS THAN ZERO*

*\*/*

IF ID\_FLEX\_NUM=50430 THEN

*/\**

*BLOCK FOR QUERY ALREADY EXIST PERCENTAGE OF FOUND NOMEEI*

*\*/*

DECLARE

BEGIN

*/\* Formatted on 27-Dec-2017 3:19:14 PM (QP5 v5.139.911.3011) \*/*

SELECT NVL (SUM (S.SEGMENT6), 0) SHARE\_PERCT

INTO FL\_SHARE\_PERCENTAGE

FROM PER\_PERSON\_ANALYSES PP, PER\_ANALYSIS\_CRITERIA S

WHERE 1 = 1

AND S.ID\_FLEX\_NUM = PP.ID\_FLEX\_NUM

AND S.ANALYSIS\_CRITERIA\_ID = PP.ANALYSIS\_CRITERIA\_ID

AND S.SEGMENT6 IS NOT NULL

AND PP.ID\_FLEX\_NUM = 50430

AND PP.PERSON\_ID = FL\_PERSONID;

*----->*

FL\_SHARE\_PERCENTAGE:=FL\_SHARE\_PERCENTAGE+user\_segment6;

END;

IF FL\_SHARE\_PERCENTAGE>100 THEN

error\_message:='Share Percentage Cannot must between 0 to 100 % .' ||FL\_PERSONID;

return false;

ELSE

return TRUE;

END IF;

ELSE

return TRUE;

END IF;

END validate;

END FND\_FLEX\_PLSQL;

/