

```

/*-----
-- script : verify if the fk columns are already indexed or not
--         the purpose of this script is to avoid creating redundant indexes
--         it has been defined to check FKs with up to 3 columns but can easily be updated to
--         handle Fks with more than 3 columns
--         when checking existing indexes for FK with one column then you can simply supply 'none'
--         without cotes (') for the remaining columns when asked by the script to do so
--
-- author   : Mohamed Hourri
-- date     : december 2010
-- version  : v1.0 --> initial creation
--           : v1.1 --> exclude the bitmap index type from the list of existing indexes
-- Example 1 sql.world> start index_fk
--                 Enter value for m_table: t1
--                 Enter value for m_column: id
--                 Enter value for m_column2: none
--
--                 table_name      index_name      column_nam  column_pos
--                 -----
--                 t1              ind_usr_ni   id          1
--
-- the output says that for the table t1 there exists already an index named ind_usr_ni starting
-- with the column id that you want to use as FK. So you don't need to create an extra index
-- to cover the locking threat of non indexed foreign keys
--
-- Example 2 sql.world> start index_fk
--                 Enter value for m_table: t2
--                 Enter value for m_column : col1
--                 Enter value for m_column2: col2
--                 Enter value for m_column3: none
--
--                 table_name      index_name      column_nam  column_pos
--                 -----
--                 t2              t2_usr_ni     col1        1
--
-- the output says that for the table t2 it exists an index named t2_usr_ni
-- starting with the column col1 but the script did not found an index
-- starting by (col1, col2) or (col2, col1).
-- So in this situation you need to create an extra index covering the FK and starting with
-- (col1, col2) or (col2, col1) to cover the locking threat of non indexed foreign keys
--
-- Example 3 sql.world> start index_fk
--                 Enter value for m_table: t3
--                 Enter value for m_column: col1
--                 Enter value for m_column2: col2

```

```

--                               Enter value for m_column3: none
-- table_name                    index_name      column_name  column_pos
-----
-- t3                            t3_ni_1    col0         1
-- t3                            t3_ni_1    col1         2
-- t3                            t3_ni_1    col2         3
-- in this situation there exist an index name t3_ni_1 having (col1, col2)
-- as indexed columns but they are at position 2 and 3 respectively. However in order to
-- cover a FK the index should have (col1, col2) at position 1 or 2 or 2 and 1 respectively
-- Hence it is necessary in this case to create an extra index to cover the FK
-----*/
define m_table_name      = &m_table
define m_column_name     = &m_column
define m_column_name2   = &m_column2
define m_column_name3   = &m_column3

spool index_col_fk.log
set verify off
set linesize 100
select substr(uc1.table_name,1,25) table_name
       ,substr(uc1.index_name,1,30) index_name
       ,substr(uc1.column_name,1,10) column_name
       ,uc1.column_position      column_pos
from user_ind_columns uc1
where uc1.table_name = upper('&m_table_name')
and uc1.column_name in (upper('&m_column_name')
                      ,upper('&m_column_name2')
                      ,upper('&m_column_name3')
                      )
and not exists (select null
               from user_indexes ui
               where ui.index_name = uc1.index_name
               and ui.index_type = 'BITMAP'
               )
order by
       uc1.index_name
       ,uc1.column_position
       ,uc1.column_position
;
spool off

```