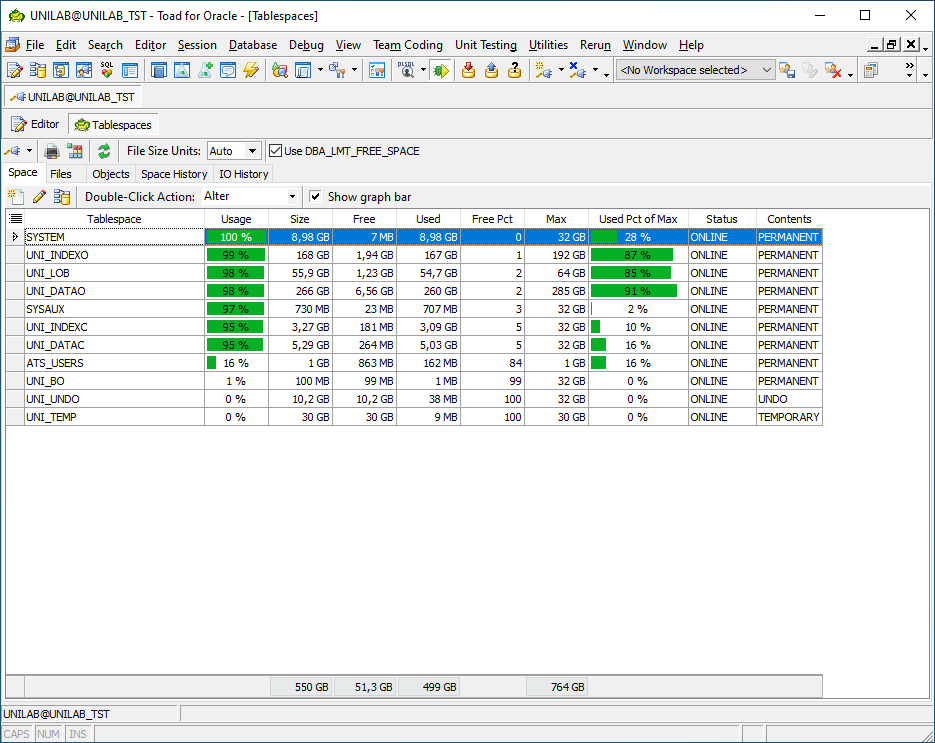
Start TOAD

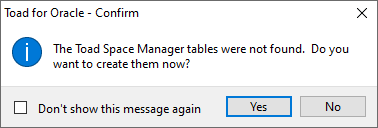
Maak connectie met SYSTEM of schema-owner UNILAB/INTERSPC

Ga via hoofdmenu [DATABASE] + [ADMINISTER] + [TABLESPACES]

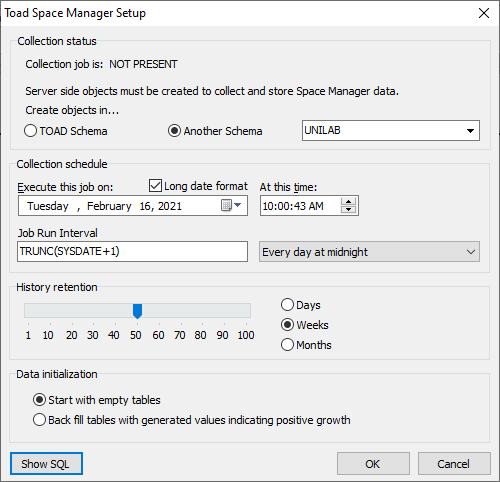


Klik in midden (net boven tablespace-overzicht) op tabblad [SPACE HISTORY]

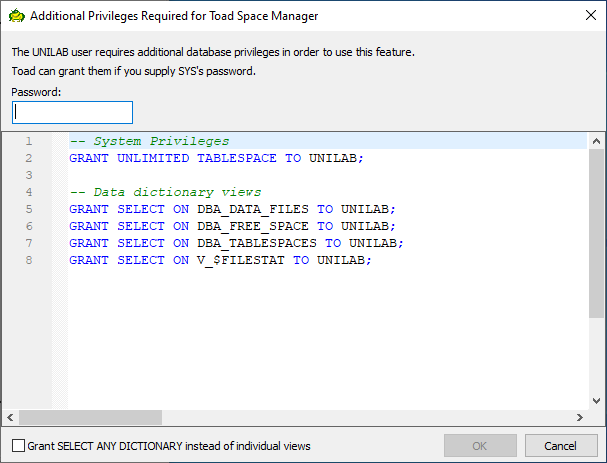
Bij de eerste keer benaderen van dit overzicht krijg je onderstaande melding:



Kik hier [yes]



Als je hier op [SHOW SQL] klikt krijg je eerst volgende melding:



-- System Privileges

GRANT UNLIMITED TABLESPACE TO UNILAB;

-- Data dictionary views

GRANT SELECT ON DBA\_DATA\_FILES TO UNILAB;

GRANT SELECT ON DBA\_FREE\_SPACE TO UNILAB;

GRANT SELECT ON DBA\_TABLESPACES TO UNILAB;

GRANT SELECT ON V\_$FILESTAT TO UNILAB;

Als je vinkje [grant select ANY DICTIONARY instead of inidividual views] aanzet:

-- System Privileges + Data dictionary views:

GRANT UNLIMITED TABLESPACE TO UNILAB;

GRANT SELECT ANY DICTIONARY TO UNILAB;

--

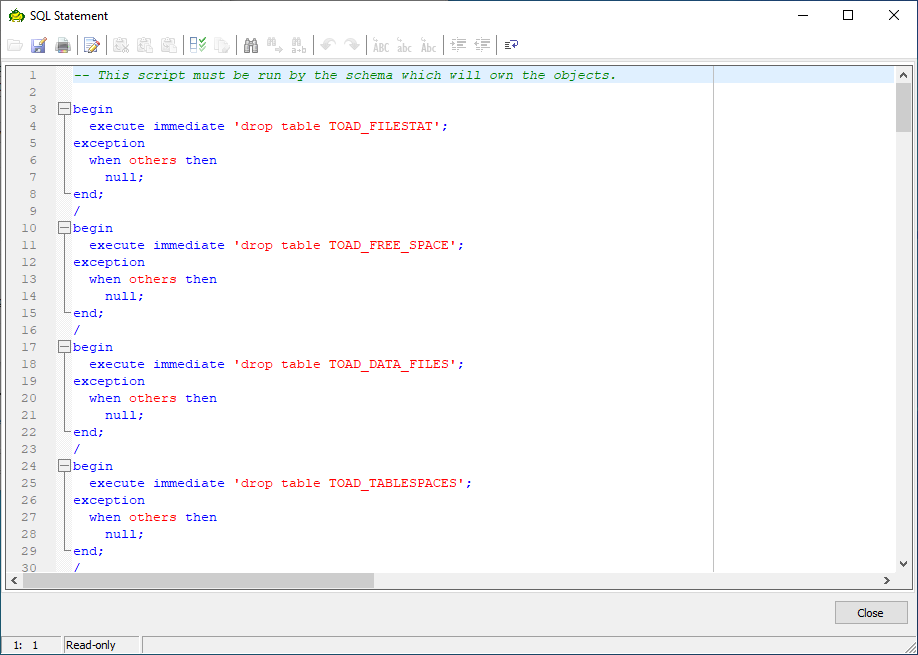
GRANT UNLIMITED TABLESPACE TO INTERSPC;

GRANT SELECT ANY DICTIONARY TO INTERSPC;

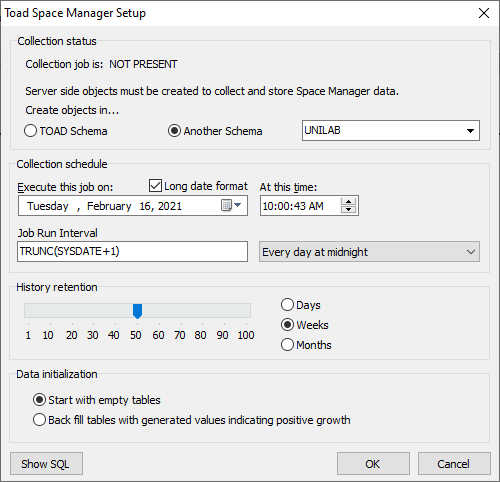
Geef UNILAB-user deze rechten mbv SYS-user !

Vul vervolgens het SYS-password in en druk op [OK]

En krijgen we het SQL-statement te zien wat gedraaid gaat worden als we doorgaan:



-- This script must be run by the schema which will own the objects.



Druk nu [OK]

BIJLAGE: toad-script tbv aanmaken objecten

begin

execute immediate 'drop table TOAD\_FILESTAT';

exception

when others then

null;

end;

/

begin

execute immediate 'drop table TOAD\_FREE\_SPACE';

exception

when others then

null;

end;

/

begin

execute immediate 'drop table TOAD\_DATA\_FILES';

exception

when others then

null;

end;

/

begin

execute immediate 'drop table TOAD\_TABLESPACES';

exception

when others then

null;

end;

/

begin

execute immediate 'drop table TOAD\_REF';

exception

when others then

null;

end;

/

CREATE TABLE TOAD\_REF (Name Varchar2(30), Value Number);

INSERT INTO TOAD\_REF VALUES ('Space Manager History Limit', 350);

INSERT INTO TOAD\_REF VALUES ('Space Manager History Unit', 1);

INSERT INTO TOAD\_REF VALUES ('Space Manager JobID', -1);

CREATE TABLE TOAD\_TABLESPACES (

TABLESPACE\_NAME VARCHAR2(30) NOT NULL,

MON\_DATE DATE NOT NULL,

CONSTRAINT TOAD\_TABLESPACES\_PK PRIMARY KEY (TABLESPACE\_NAME, MON\_DATE)

USING INDEX PCTFREE 5)

PCTFREE 5 PCTUSED 94;

CREATE TABLE TOAD\_DATA\_FILES (

FILE\_ID NUMBER NOT NULL,

MON\_DATE DATE NOT NULL,

TABLESPACE\_NAME VARCHAR2(30) NOT NULL,

FILE\_NAME VARCHAR2(257) NOT NULL,

BYTES NUMBER,

CONSTRAINT TOAD\_DATA\_FILES\_PK PRIMARY KEY (FILE\_ID, MON\_DATE)

USING INDEX PCTFREE 5,

CONSTRAINT TOAD\_DATA\_FILES\_FK FOREIGN KEY (TABLESPACE\_NAME, MON\_DATE)

REFERENCES TOAD\_TABLESPACES ON DELETE CASCADE)

PCTFREE 5 PCTUSED 94;

CREATE UNIQUE INDEX TOAD\_DATA\_FILES\_UK ON TOAD\_DATA\_FILES (TABLESPACE\_NAME, FILE\_NAME, MON\_DATE) PCTFREE 5;

CREATE TABLE TOAD\_FILESTAT (

FILE\_ID NUMBER NOT NULL,

MON\_DATE DATE NOT NULL,

PHYRDS CHAR(18) NOT NULL,

PHYWRTS CHAR(18) NOT NULL,

PHYBLKRD CHAR(18) NOT NULL,

PHYBLKWRT CHAR(18) NOT NULL,

READTIM CHAR(18) NOT NULL,

WRITETIM CHAR(18) NOT NULL,

CONSTRAINT TOAD\_FILESTAT\_PK PRIMARY KEY (FILE\_ID, MON\_DATE)

USING INDEX PCTFREE 5,

CONSTRAINT TOAD\_FILESTAT\_FK FOREIGN KEY (FILE\_ID, MON\_DATE)

REFERENCES TOAD\_DATA\_FILES ON DELETE CASCADE

) PCTFREE 5 PCTUSED 94;

CREATE TABLE TOAD\_FREE\_SPACE (

FILE\_ID NUMBER NOT NULL,

MON\_DATE DATE NOT NULL,

BYTES NUMBER NOT NULL,

CONSTRAINT TOAD\_FREE\_SPACE\_PK PRIMARY KEY (FILE\_ID, MON\_DATE)

USING INDEX PCTFREE 5,

CONSTRAINT TOAD\_FREE\_SPACE\_FK FOREIGN KEY (FILE\_ID, MON\_DATE)

REFERENCES TOAD\_DATA\_FILES ON DELETE CASCADE

) PCTFREE 5 PCTUSED 94;

CREATE OR REPLACE FORCE VIEW TOAD\_SPACE

(TABLESPACE, FREE\_MEGS, USED\_MEGS, ALLOC\_MEGS, PCT\_FREE, PCT\_USED)

AS

SELECT tdf.tablespace\_name,

ROUND(SUM(NVL(tfs.bytes,0))/1048576),

ROUND((SUM(tdf.bytes)-SUM(NVL(tfs.bytes,0)))/1048576),

SUM(tdf.BYTES)/1048576,

ROUND(SUM(NVL(tfs.bytes,0))/SUM(tdf.bytes),2),

1-ROUND(SUM(NVL(tfs.bytes,0))/SUM(tdf.bytes),2)

FROM toad\_data\_files tdf, toad\_free\_space tfs

WHERE tdf.file\_id = tfs.file\_id (+)

AND tdf.mon\_date = tfs.mon\_date (+)

AND tdf.mon\_date <= SYSDATE

GROUP BY tdf.tablespace\_name;

CREATE OR REPLACE FORCE VIEW TOAD\_SPACE\_HIST

(file\_id, tablespace, mdate, usedspace)

AS

SELECT a.file\_id, a.tsp, a.mdate, GREATEST(ROUND(SUM(a.space)/1048576), 0)

FROM (SELECT tdf.file\_id file\_id,

tdf.tablespace\_name tsp,

tdf.mon\_date mdate,

SUM(tdf.bytes) - SUM(NVL(tfs.bytes,0)) space

FROM toad\_data\_files tdf, toad\_free\_space tfs

WHERE tdf.file\_id = tfs.file\_id (+)

AND tdf.mon\_date = tfs.mon\_date (+)

GROUP BY tdf.file\_id,

tdf.tablespace\_name,

tdf.mon\_date ) a

GROUP BY file\_id, tsp, mdate;

CREATE OR REPLACE PUBLIC SYNONYM TOAD\_FILESTAT FOR TOAD\_FILESTAT;

CREATE OR REPLACE PUBLIC SYNONYM TOAD\_FREE\_SPACE FOR TOAD\_FREE\_SPACE;

CREATE OR REPLACE PUBLIC SYNONYM TOAD\_DATA\_FILES FOR TOAD\_DATA\_FILES;

CREATE OR REPLACE PUBLIC SYNONYM TOAD\_TABLESPACES FOR TOAD\_TABLESPACES;

CREATE OR REPLACE PUBLIC SYNONYM TOAD\_REF FOR TOAD\_REF;

CREATE OR REPLACE PUBLIC SYNONYM TOAD\_SPACE\_HIST FOR TOAD\_SPACE\_HIST;

CREATE OR REPLACE PUBLIC SYNONYM TOAD\_SPACE FOR TOAD\_SPACE;

GRANT SELECT, INSERT, UPDATE, DELETE ON TOAD\_FILESTAT TO PUBLIC;

GRANT SELECT, INSERT, UPDATE, DELETE ON TOAD\_FREE\_SPACE TO PUBLIC;

GRANT SELECT, INSERT, UPDATE, DELETE ON TOAD\_DATA\_FILES TO PUBLIC;

GRANT SELECT, INSERT, UPDATE, DELETE ON TOAD\_TABLESPACES TO PUBLIC;

GRANT SELECT, INSERT, UPDATE, DELETE ON TOAD\_SPACE\_HIST TO PUBLIC;

GRANT SELECT, INSERT, UPDATE, DELETE ON TOAD\_SPACE TO PUBLIC;

GRANT SELECT, INSERT, UPDATE, DELETE ON TOAD\_REF TO PUBLIC;

CREATE OR REPLACE PACKAGE TOAD\_SPACEMAN IS

PROCEDURE CAPTURE;

PROCEDURE RESET;

END;

/

CREATE OR REPLACE PACKAGE BODY TOAD\_SPACEMAN IS

PROCEDURE RESET AS

BEGIN

/\* DELETE DATA FROM ALL TABLES \*/

DELETE FROM TOAD\_TABLESPACES;

/\* LOAD REAL DATA FOR TODAY \*/

TOAD\_SPACEMAN.CAPTURE;

/\* LOAD DUMMY DATA FOR 60 DAYS \*/

FOR I IN 1..60 LOOP

INSERT INTO TOAD\_TABLESPACES

SELECT TABLESPACE\_NAME, TRUNC(MON\_DATE-I)

FROM TOAD\_TABLESPACES WHERE TRUNC(MON\_DATE) = TRUNC(SYSDATE);

INSERT INTO TOAD\_DATA\_FILES

SELECT FILE\_ID, TRUNC(MON\_DATE-I), TABLESPACE\_NAME, FILE\_NAME, ROUND(BYTES\*(1-I/100))

FROM TOAD\_DATA\_FILES WHERE TRUNC(MON\_DATE) = TRUNC(SYSDATE);

INSERT INTO TOAD\_FREE\_SPACE

SELECT FILE\_ID, TRUNC(MON\_DATE-I), ROUND(BYTES\*(1+I/100))

FROM TOAD\_FREE\_SPACE WHERE TRUNC(MON\_DATE) = TRUNC(SYSDATE);

INSERT INTO TOAD\_FILESTAT

SELECT FILE\_ID, TRUNC(MON\_DATE-I), PHYRDS, PHYWRTS,

PHYBLKRD, PHYBLKWRT, READTIM, WRITETIM

FROM TOAD\_FILESTAT WHERE TRUNC(MON\_DATE) = TRUNC(SYSDATE);

END LOOP;

COMMIT;

END RESET;

PROCEDURE CAPTURE AS

V\_MON\_DATE DATE := NULL;

V\_ROW\_SIZE INTEGER := 100000;

NumDaysToRemove NUMBER;

CURSOR History\_Cur IS

SELECT Value

FROM TOAD\_REF

WHERE Name = 'Space Manager History Limit';

BEGIN

/\* SAVE TODAYS DATE \*/

V\_MON\_DATE := SYSDATE; -- removed Trunc() to allow interval less than one day

/\* RETRIEVE THE NUMBER OF DAYS TO REMOVE \*/

OPEN History\_Cur;

FETCH History\_Cur INTO NumDaysToRemove;

/\* DELETE DATA OLDER THAN USER SPECIFIED DAYS \*/

DELETE FROM TOAD\_TABLESPACES WHERE MON\_DATE < V\_MON\_DATE - NumDaysToRemove;

/\* DELETE PRE-EXISTING DATA FOR CURRENT DATE AND TIME \*/

DELETE FROM TOAD\_TABLESPACES WHERE MON\_DATE = V\_MON\_DATE;

/\* DELETE DATA FOR NONEXISTENT TABLESPACES \*/

DELETE FROM TOAD\_TABLESPACES X

WHERE NOT EXISTS (SELECT 1 FROM DBA\_TABLESPACES D

WHERE X.TABLESPACE\_NAME = D.TABLESPACE\_NAME);

--/\* DELETE DATA FOR NONEXISTENT DATA FILES \*/

--DELETE FROM TOAD\_DATA\_FILES X

-- WHERE NOT EXISTS (SELECT 1 FROM DBA\_DATA\_FILES D

-- WHERE X.TABLESPACE\_NAME = D.TABLESPACE\_NAME

-- AND X.FILE\_NAME = D.FILE\_NAME);

/\* INSERT NEW TABLESPACES FOUND \*/

INSERT INTO TOAD\_TABLESPACES

SELECT TABLESPACE\_NAME, V\_MON\_DATE

FROM DBA\_TABLESPACES;

/\* INSERT NEW DATA FILES FOUND \*/

INSERT INTO TOAD\_DATA\_FILES

SELECT FILE\_ID, V\_MON\_DATE, TABLESPACE\_NAME, FILE\_NAME, BYTES

FROM DBA\_DATA\_FILES;

/\* COLLECT CURRENT FREE SPACE DATA \*/

INSERT INTO TOAD\_FREE\_SPACE

SELECT FILE\_ID, V\_MON\_DATE, SUM(BYTES)

FROM DBA\_FREE\_SPACE

GROUP BY FILE\_ID, TRUNC(SYSDATE);

/\* COLLECT CURRENT IO RATE DATA \*/

INSERT INTO TOAD\_FILESTAT

SELECT FILE#, V\_MON\_DATE, PHYRDS, PHYWRTS,

PHYBLKRD, PHYBLKWRT, READTIM, WRITETIM

FROM V$FILESTAT;

COMMIT;

END CAPTURE;

END TOAD\_SPACEMAN;

/

CREATE OR REPLACE PUBLIC SYNONYM TOAD\_SPACEMAN FOR TOAD\_SPACEMAN;

GRANT EXECUTE ON TOAD\_SPACEMAN TO PUBLIC;

UPDATE TOAD\_REF SET Value = 1

WHERE Name = 'Space Manager History Unit';

UPDATE TOAD\_REF SET Value = 350

WHERE Name = 'Space Manager History Limit';

DECLARE

X NUMBER;

BEGIN

**SYS.DBMS\_JOB.SUBMIT**

(

job => X

,what => 'BEGIN

TOAD\_SPACEMAN.CAPTURE;

END;'

,next\_date => to\_date('16/02/2021 10:00:43','dd/mm/yyyy hh24:mi:ss')

,interval => 'TRUNC(SYSDATE+1)'

,no\_parse => TRUE

);

UPDATE TOAD\_REF SET Value = to\_char(x)

WHERE Name = 'Space Manager JobID';

END;

/

COMMIT;