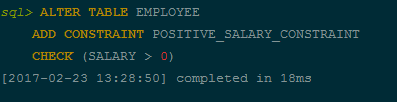
Question 8

1. For our first constraint, we want to make sure no negative numbers are recorded as the salaries of employees. Hence we will alter our table for EMPLOYEE such that it checks for SALARY to be > 0 on every entry created!

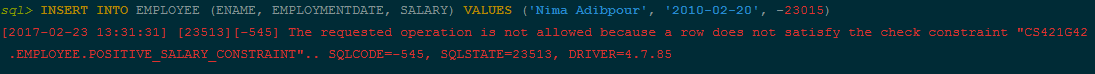
**ALTER TABLE** EMPLOYEE  
 **ADD CONSTRAINT** POSITIVE\_SALARY\_CONSTRAINT  
 **CHECK** (**SALARY** > 0);

OUTPUT:



And here is our attempt at violating the constraint:

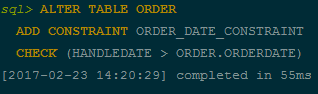
**INSERT INTO** EMPLOYEE (**ENAME**, **EMPLOYMENTDATE**, **SALARY**) **VALUES** (**'Nima Adibpour'**, **'2010-02-20'**, -23015);

And the DB2 response:

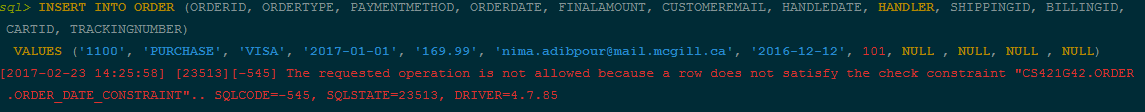
1. In our second constraint, we want to make sure that the orders are handled after they are actually placed! This is a check constraint that makes sure handle date is after order date, in the order table.

**ALTER TABLE ORDER  
 ADD CONSTRAINT** ORDER\_DATE\_CONSTRAINT  
 **CHECK** (**HANDLEDATE** > ORDER.**ORDERDATE**)

OUTPUT:



And here is our attempt at violating the constraint:

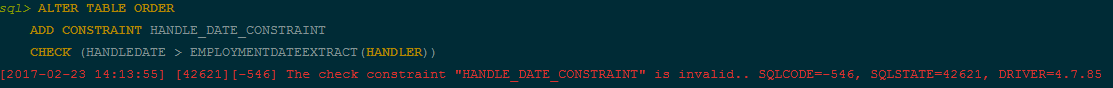


So as visible, order date was 2017-01-01 here but handle date was 2016-12-12!

1. ***DO NOT CONSIDER THIS FOR MARKING:*** This is another constraint that seemed very interesting for me so I wanted to show it to you but sadly it does not work yet!

**CREATE FUNCTION** *EMPLOYMENTDATEEXTRACT*(myEID **INT**)  
 **RETURNS DATE  
 BEGIN  
 RETURN** (**SELECT EMPLOYMENTDATE from** EMPLOYEE **WHERE** EMPLOYEE.**EID** = myEID);  
 **END**;

**ALTER TABLE ORDER  
 ADD CONSTRAINT** HANDLE\_DATE\_CONSTRAINT  
 **CHECK** (**HANDLEDATE** > *EMPLOYMENTDATEEXTRACT*(**HANDLER**));

I tried running this, the function I created works, but the constraint could not be placed. Here’s the output error which since I couldn’t fix it within the given time I gave up on!