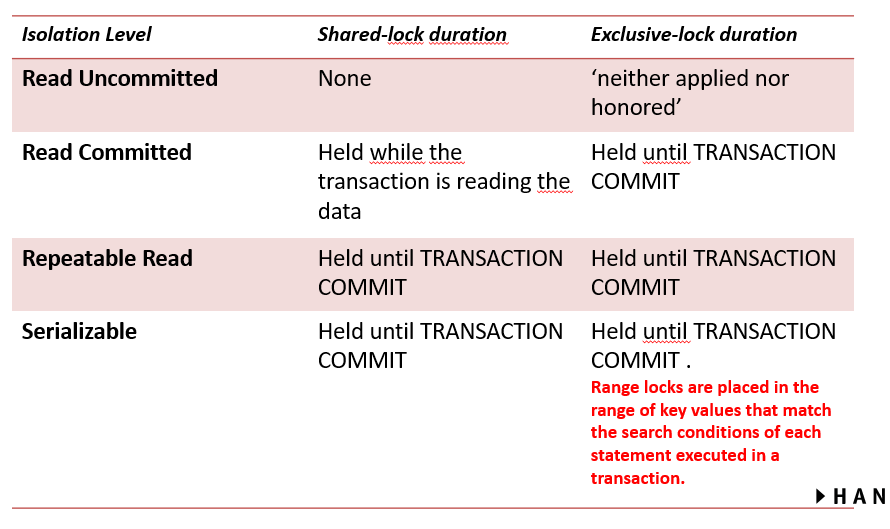
Choose **two** of your procedural **constraints**. One which has no problems with non-repeatable read or phantoms, in a multi-user environment under the default isolation level (READ COMMITTED) and one which does.

Explain your choices by giving scenarios with two transactions that illustrate why it can or can’t go wrong. Add a success scenario with an isolation level that solves the problematic one.

For every scenario describe what kind of locks are acquired (e.g. s-locks and x-locks), when, why and for how long.



**No problems with non-repeatable reads/phantoms**

|  |  |
| --- | --- |
| **Connection 1 – Read commited** | **Connection 2 – Read commited** |
| go  ALTER PROCEDURE usp\_insertReg  (  @stud numeric(4),  @course varchar(6),  @starts date,  @eval numeric(1)  )  AS  BEGIN TRY  if((select count(\*) from reg where course = @course and starts = @starts) >= 6)  update offr set status = 'CONF' where course = @course and starts = @starts  END TRY  BEGIN CATCH  ;THROW  END CATCH  go |  |
|  |  |

**Problem with non-repeatable reads/phantoms**

|  |  |
| --- | --- |
| **Connection 1 – Read commited** | **Connection 2 – Read commited** |
|  |  |
|  |  |

**Solution**

|  |  |
| --- | --- |
| **Connection 1 – Read commited** | **Connection 2 – Read commited** |
|  |  |
|  |  |