As we replicate the implications of the Isolation levels, we will use the Member.membrNumber as the field that display consistency/inconsistency

**Isolation Level:** ReadUnCommitted

**DIRTY READ**

Set memberNumber to 1.

Start Transaction A [in TransactionService], set memberNumber to 12

Start **NEW Transaction B** [in ReadUnCommittedService] read memberNumber = 12 \*\*\*Dirty

END Transaction B

ROLLBACK Transaction A

read memberNumber = 1

**Isolation Level:** ReadCommitted

**NO DIRTY READ**

Start **Transaction A** [in TransactionService], set memberNumber to 12

Start **NEW** **Transaction B** [in ReadCommittedService] read memberNumber = 1 \*\*\* GOOD DATA

END Transaction B

ROLLBACK Transaction A

read memberNumber = 1

**Isolation Level:** ReadCommitted

**NON REPEATABLE READ**

Start **Transaction A** [in ReadCommittedService], read memberNumber = 1

Start **NEW** **Transaction B** [in TransactionService] set memberNumber = 56

END Transaction B

**Transaction A** [in ReadCommittedService], read memberNumber = 56 \*\* should be 1

END Transaction A

read memberNumber = 56

**Isolation Level:** repeatable\_read

**REPEATABLE READ**

Start **Transaction A** [in RepeatableReadService], read memberNumber = 56

Start **NEW** **Transaction B** [in TransactionService] set memberNumber = 100

END Transaction B

**Transaction A** [in RepeatableReadService], read memberNumber = 56 \*\* GOOD DATA

END Transaction A

read memberNumber = 100

**NOTE reproduced for READCOMMITED but could NOT for REPEATABLE\_READ**

**Isolation Level:** repeatable\_read

**PHANTOM READ**

Start **Transaction A** [in RepeatableReadService], count = 3

Start **NEW** **Transaction B** [in TransactionService] add member count = 4

END Transaction B

**Transaction A** [in RepeatableReadService], count = 4 IF it FAILS

END Transaction A