CS 338 Fun Assignment #5 - Solutions

Question 1.

(a) Solution: READ COMMITTED. 1 read 1 write, no dirty reads.
(b) Solution: REPEATABLE READ. 2 reads, read only, no dirty reads; Phantom allowed as not writing.
(c) Solution: SERIALIZABLE. 3 reads, 1 write, no dirty reads.

Question 2

(a) Solution:
An **unrepeatable read** occurs when Transaction reads the same row twice and obtains different values because another committed transaction modified that row between reads (e.g., READ COMMITTED allows this).

A **phantom read** occurs when Transaction executes the same `SELECT` that retrieves a *set* of rows twice and finds that new rows have been inserted or existing rows ahs been deleted (e.g., REPEATABLE READ allows this, SERIALIZABLE blocks it).

(b) Solution:
Benefits:

- Speeds up foreign-key joins without requiring full table scans (q3 part b is an example).
- Fast look up for selective queries (binary search with $\log(\#row)$ time).

Problems:

- Every INSERT, UPDATE, or DELETE triggers index maintenance.

Question 3.

(a) Solution:

Minimum isolation level: SERIALIZABLE: 2 read 2 write, no phantoms allowed. 1 read in `SELECT`, 1 read and 1 write in `UPDATE`, 1 write in `INSERT`

(b) Solution:

T1:

- `CREATE UNIQUE INDEX idx_book_isbn ON Book(isbn);`

T2:

- `CREATE INDEX idx_checkout_composite ON Checkout(member_id, returned_date, isbn);`
- Database can find all unreturned checkouts for member 12345 using only the index then find `returned_date IS NULL`, then joins with Book using the ISBN values directly from the index.