#1b — This is legal because `NULL` FKs aren't checked.

#1d — This not legal. `Title` is omitted so it would get its default value, which is `NULL`, but that violates `Title`'s `NOT NULL` constraint.

#1g — This is legal because a `CHECK` condition evaluating to `UNKNOWN` (due to `NULL`) passes.

#2a — Only one row is changed. Although three rows match the condition, two of them already have the publisher Eos so they are not changed.

#2b — One row in `PUBLISHER` and two in `BOOK` are changed (updates are cascaded).

#2c — This fails in MySQL though it would succeed in standard SQL. MySQL updates the rows one at a time, in key order, and checks the primary key constraint after each. Adding `ORDER BY Book_id DESC` would fix it.

#3e — Always beware the potential for `NULL` to throw a wrench into things. The `SELECT` result includes `NULL` (because there's a book with no publisher), which means the `NOT IN` comparison results in `UNKNOWN`. `WHERE` only matches rows for which the condition is `TRUE`, so no rows are matched. Use `NOT EXISTS` instead or exclude `NULL` publishers from the subquery results.

#3g — `= NULL` always evaluates to `UNKNOWN`. Use `IS NULL` instead.

#4b — Only `Book_id`, `Branch_id`, `Card_no`, and `Date_out` need to be inserted: `Loan_id` is generated automatically and both `Due_date` and `Returned_date` should be set to their defaults.

#5d, #5f — The question references "Wood Library", not branch 2. The query should work no matter what branch ID Wood Library has.

#6b — The question references a specific book title, not a book ID. The query should work no matter what book ID that title has.

#6b — The `ON DELETE RESTRICT` on `BOOK_LOANS` and `BOOK_COPIES` means the book must be deleted from those first, before removing it from `BOOK` itself.