Solutions

Marking: **M1** a correct method, **A1** the correct answer, **B1** an independent mark.

Each answer shows the steps, then the **result**.

2.1

update: a lecturer moves office and only some rows are changed

insert: a new lecturer's office cannot be stored until someone enrols on their course

delete: deleting the last enrolment on a course loses its lecturer's office

2.2

1NF: one cell holds two values

a table `phone(sid, number)` with one row per number

2.3

A^+: start with A; add B; then C

$A^+ = \{A, B, C\}$

D is not determined by anything, so the key is AD

2.4

no

`tutor → course`, but `tutor` is not a superkey (it is in 3NF only, since `course` is part of a key)

3.1

(a) `sid → sname`; `cid → ctitle, lecturer`; `lecturer → office`; `(sid, cid) → mark`

(b) `student(sid, sname)`, `enrol(sid, cid, mark)`; `course(cid, ctitle, lecturer, office)`

(c) `course(cid, ctitle, lecturer)` and `lecturer(lecturer, office)`

(d) every split is on a column that is the key of one part, so joining on it rebuilds each original row exactly

3.2

(a) $4 \times 150 =$ **600** rows; **1** row in `lecturer`

(b) the lecturer has two different offices in the data, and queries disagree

(c) every row needs a `sid` as part of its key, so the course's title and lecturer cannot be stored until a student enrols

3.3

(a) reads need no joins, so they are fast; the same facts are stored twice, so the copies can disagree and use more space

(b) rebuild it on a schedule, e.g. a materialised view refreshed hourly; or update it with triggers on every change, which is fresh but slows writes

(c) $10^7 \times 200$ B = **2 GB**

(d) a materialised view refreshed every hour: 5000 fast reads a day justify 2 GB, the hour-old data is accepted, and the normalised tables stay the single source of truth for writes

exercises.lol exercises.com.cn