Task 2:

SCHEMA #1: ROOM(bldg-number, room-number, room-area)

primary key = (bldg-number, room-number)
Functional Dependencies:
bldg-number, room-number → room-area

Minimal keys:
bldg-number, room-number

Current normal form:
BCNF because all LHS are superkeys

Denormalise to 1NF by adding attribute: (faculty) so that bldg-number→ faculty
Ie. new schema ROOM(bldg-number, room-number, room-area, faculty)
primary key = (bldg-number, room-number)
-Now schema is not in 2NF because non-prime faculty is partially dependent on primary key (bldg-number, room-number)

SCHEMA #2: FLIGHT(flight-number, from-city, to-city)

primary key = (flight-number)
Functional Dependencies:
flight-number→ from-city, to-city

Minimal keys:
Flight-number

Current normal form:
BCNF because LHS is superkeys

Denormalise to 2NF by adding attribute: (arrivalCityTemperature) So that to-city → arrivalCityTemperature

-2NF because no non-prime is partially dependent on primary key
-NOT in 3NF because non-prime attribute ‘arrivalCityTemperature’ is transitively dependent on primary key
Ie. flight-number→ to-city and to-city→ arrivalCityTemperature

SCHEMA #3: ORDER(order-number, supplier)

primary key = (order-number)
Functional Dependencies:
order-number→ supplier

Minimal keys:
order-number

Current normal form:
BCNF all LHS are superkeys

Denormalise to 3NF by adding attribute: (customerID) so that customerID, supplier→ order-number
NOT BCNF because customerID, supplier on LHS is not a prime attribute
But schema is in 3NF because for all functional dependencies:

-LHS is a primary key (ie. order-number→ supplier, customerID)

or

-RHS is a prime attribute (part of minimal key) (customerID, supplier→ order-number)