CHAPTER 2 – Database Requirements and ER Modeling

KEY CONCEPTS

The goal of this chapter is to provide comprehensive coverage of entity-relationship (ER) modeling - a conceptual method for visualizing and structuring user database requirements.

The following topics related to ER modeling are covered:

- Visualizing and Structuring the Database Requirements with ER Modeling
- ER Modeling Components
  - Entities
    - Regular
    - Weak
    - Associative
  - Relationships
    - Cardinality Constraints
      - Maximum Cardinality - One or Many
      - Minimum Cardinality (Participation) – Optional or Mandatory
    - Degrees:
      - Binary
      - Unary
      - Ternary
    - Maximum Cardinality-Wise
      - 1:M
      - M:N
      - 1:1
    - Identifying Relationships
    - Exact Minimum and Maximum Cardinalities
    - Relationship Roles
    - Multiple Relationships between the Same Entities
  - Attributes
    - Regular
    - Unique
    - Multiple Unique (Candidate Keys)
    - Composite
    - Composite Unique
    - Derived
    - Multivalued
    - Optional
    - Relationship Attributes

Chapter 2 - ANSWERS (Review Questions)

Q2.1 *What is the purpose of ER modeling?*

ER modeling is a conceptual database modeling method. It enables the structuring and organizing of the requirements collection process and provides a way to graphically represent the requirements.

Q2.2 *What are the basic ER modeling constructs?*

Entities (with their attributes) and relationships.

Q2.3 *What is a unique attribute?*

An attribute whose value is different for each entity instance.

Q2.4 *What is depicted by cardinality constraints?*

Cardinality constraints depict how many instances of one entity can be associated with instances of another entity

Q2.5 *What are the four possible cardinality constraints?*

- Mandatory Many
- Optional Many
- Mandatory One
- Optional One

Q2.6 *What are the three types of relationships (maximum cardinality-wise)?*

- One-to-one relationship (1:1)
- One-to-many relationship (1:M)
- Many-to-many relationship (M:N)

Q2.7 *What is a composite attribute?*

An attribute that is composed of several attributes.

Q2.8 *What are candidate keys?*

Multiple unique attributes of the same entity.

Q2.9 *What is a multivalued attribute?*

An attribute for which instances of an entity can have multiple values.

Q2.10 *What is a derived attribute?*

An attribute whose value is calculated (from stored values of other attributes and/or additional available data) and not permanently stored in a database.

Q2.11 *What is an optional attribute?*

An attribute that is allowed to not have a value

Q2.12 *How are exact minimum and maximum cardinalities depicted in a relationship?*

By pairs of numbers in parentheses placed on the relationship lines. The number closest to the open parenthesis indicates minimum cardinality. The second number, the number closest to the closed parenthesis, indicates maximum cardinality.

Q2.13 *What is a binary relationship?*

A relationship that involves two entities.

Q2.14 *What is a unary relationship?*

A relationship involving one entity in a relationship with itself.

Q2.15 *What is a weak entity?*

A construct in an ER diagram used to depict entities that do not have a unique attribute of their own.

Q2.16 *What is an associative entity?*

An ER diagram construct used as an alternative way of depicting M:N relationships.

Q2.17 *What is a ternary relationship?*

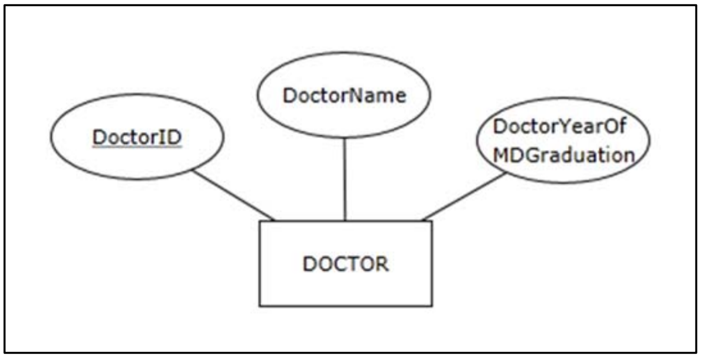
A relationship involving three entities

Chapter 2 - ANSWERS (Exercises)

E2.1 *Create an entity based on the following scenarios.*

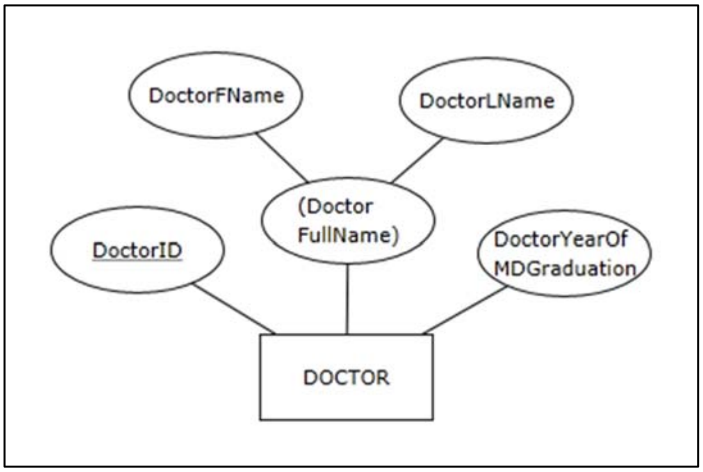
E2.1a *Create an entity based on the following requirement:*

- For each doctor, we will keep track of a DoctorID (unique), DoctorName, and DoctorYearOfMDGraduation (year of graduating from medical school).



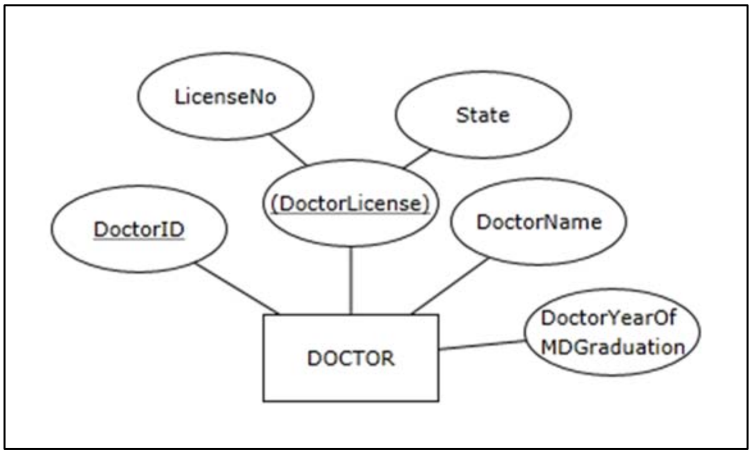
E2.1b *Create an entity based on the following requirement:*

- For each doctor, we will keep track of a DoctorID (unique), DoctorName (composed of first and last name), and DoctorYearOfMDGraduation (year of graduating from medical school).



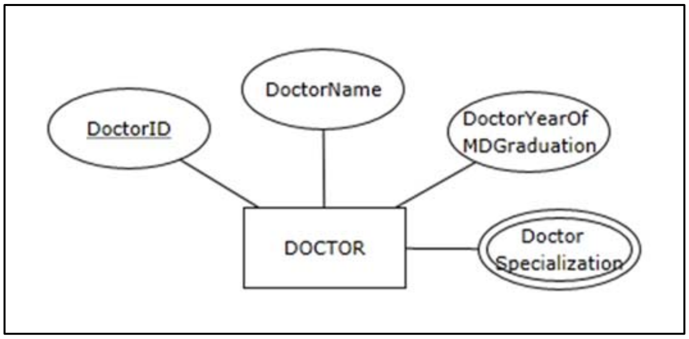
E2.1c *Create an entity based on the following requirement:*

- For each doctor, we will keep track of a DoctorID (unique), DoctorLicense (unique, composed of LicenseNo and State), DoctorName, and DoctorYearOfMDGraduation (year of graduating from medical school).



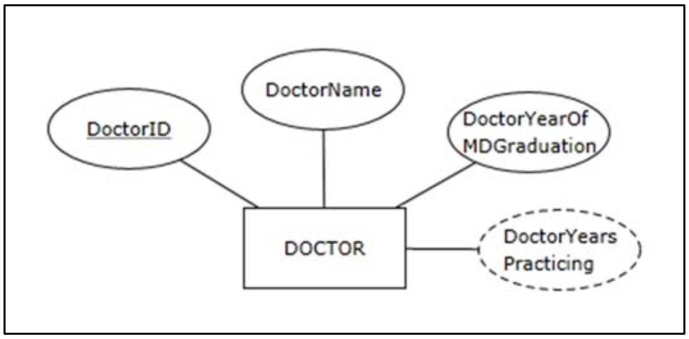
E2.1d *Create an entity based on the following requirement:*

- For each doctor, we will keep track of a DoctorID (unique), DoctorName, DoctorYearOfMDGraduation (year of graduating from medical school), and one or more specializations.



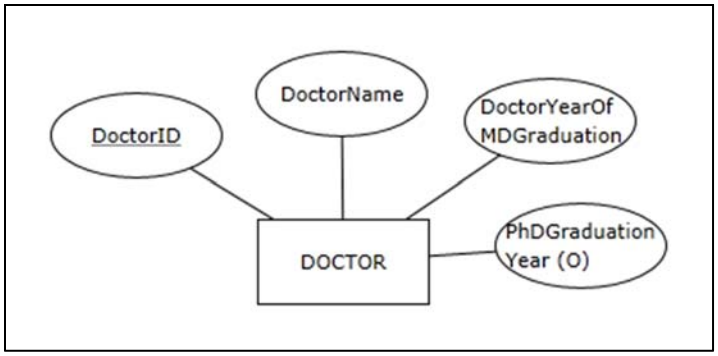
E2.1e *Create an entity based on the following requirement:*

- For each doctor, we will keep track of a DoctorID (unique), DoctorName, DoctorYearOfMDGraduation (year of graduating from medical school), and DoctorYearsPracticing (how many years the doctor has been practicing medicine).



E2.1f *Create an entity based on the following requirement:*

- For each doctor, we will keep track of a DoctorID (unique), DoctorName, DoctorYearOfMDGraduation (year of graduating from medical school), and DoctorYearOfPhDGraduation (year of acquiring a PhD degree; not every doctor has a PhD degree).

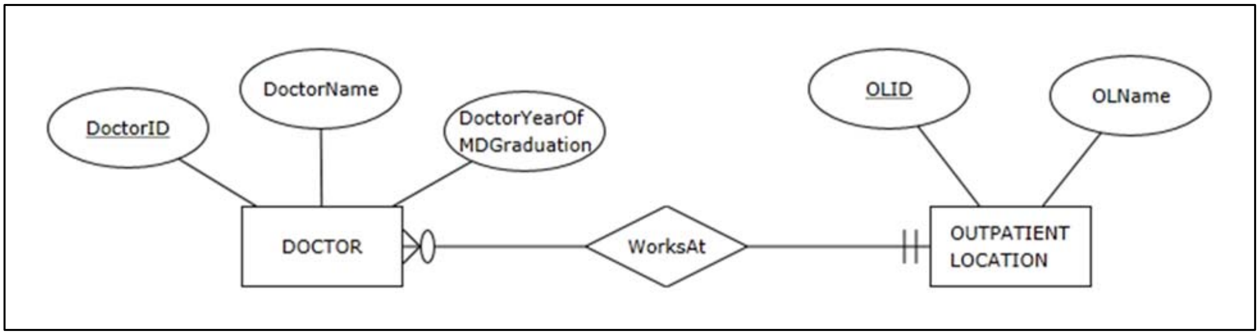


***E2.2** Create the ER diagrams for a scenario based on these requirements:*

- A health care organization keeps track of its doctors and outpatient locations.
- For each doctor it keeps track of the DoctorID (unique), DoctorName, and DoctorYearOfMDGraduation (year of graduating from medical school).
- For each outpatient location it keeps track of the OLID (unique) and OLName, and the following additional requirement:

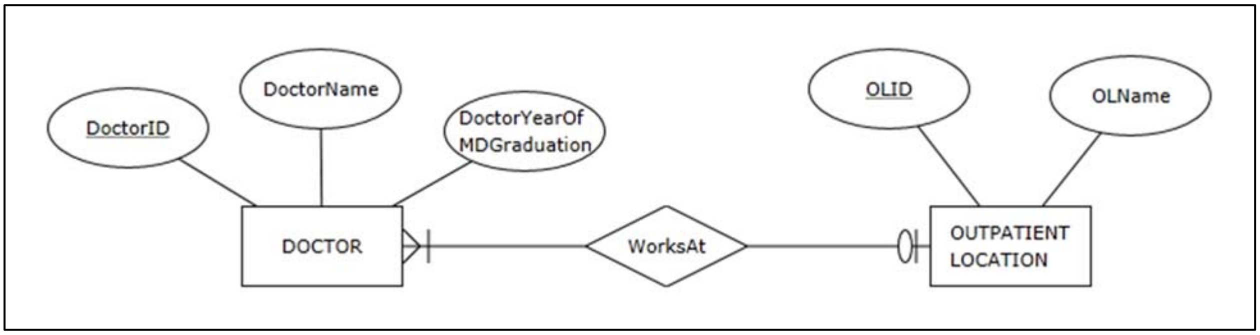
***E2.2a** Additional requirement*

- Each doctor works at exactly one outpatient location, and each outpatient location can have between none (rehab and minor health issues only) and many doctors working at it.



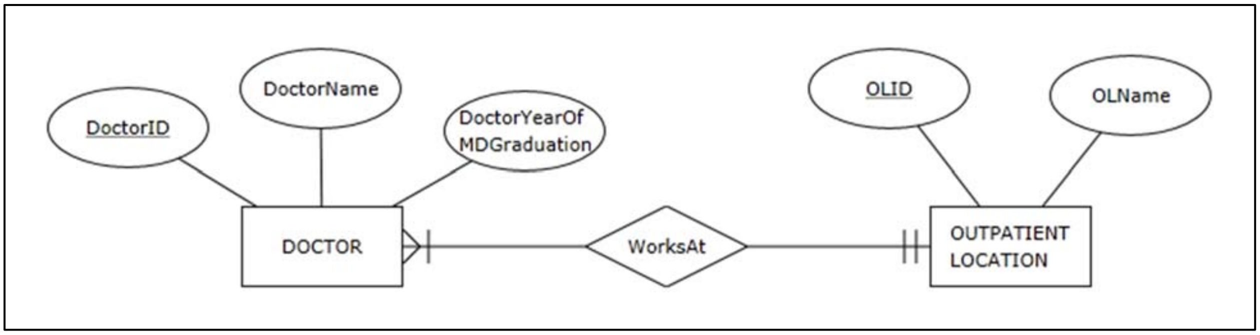
***E2.2b** Additional requirement*

- Each doctor works at either one outpatient location or at none (strictly working in the main hospital), and each outpatient location must have at least one but can have many doctors working at it.



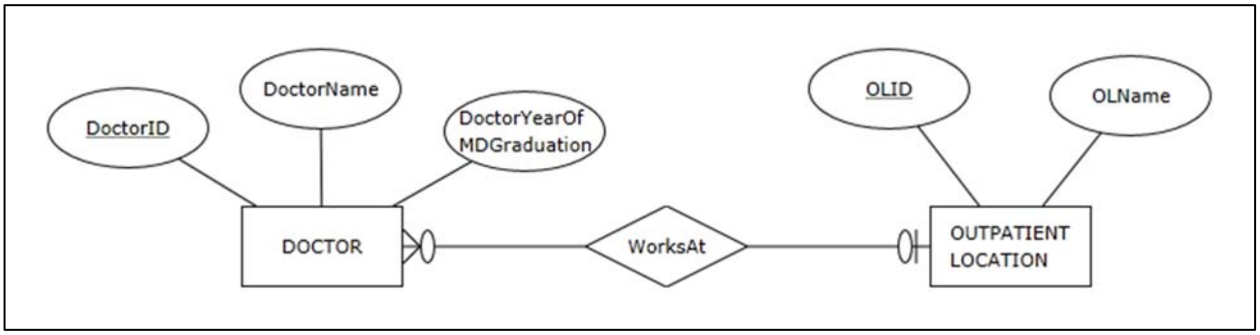
***E2.2c** Additional requirement*

- Each doctor works at exactly one outpatient location, and each outpatient location must have at least one but can have many doctors working at it.



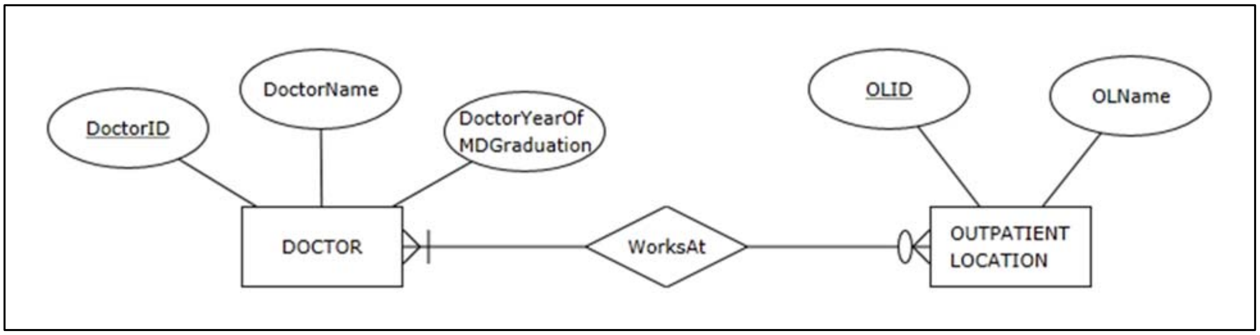
***E2.2d** Additional requirement*

- Each doctor works at either one outpatient location or at none (strictly working in the main hospital), and each outpatient location can have between none (rehab and minor health issues only) and many doctors working at it.



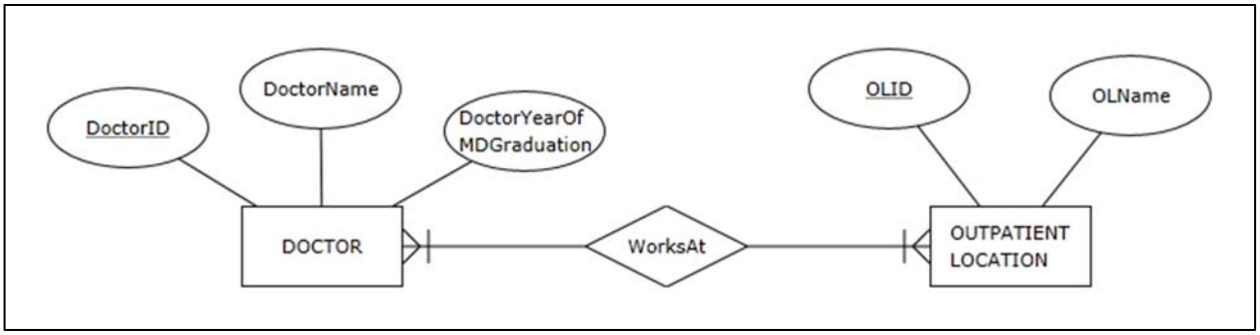
***E2.2e** Additional requirement*

- Each doctor works at between none (strictly working in the main hospital), and many outpatient locations, and each outpatient location must have at least one but can have many doctors working at it.



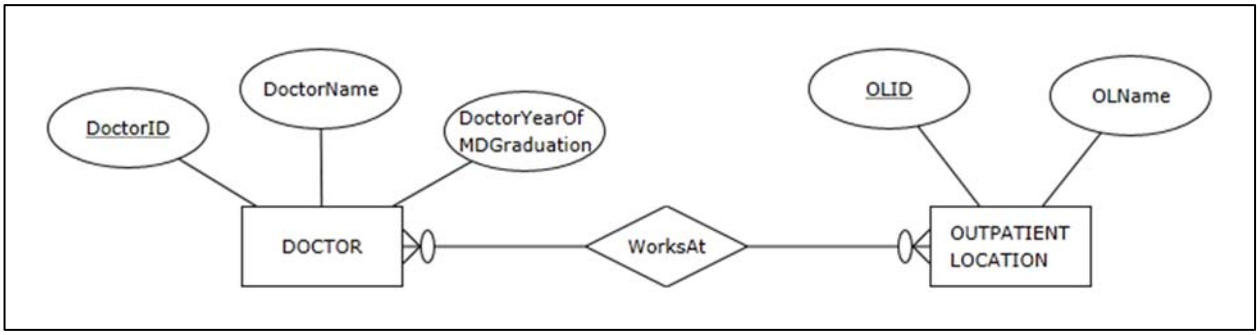
E2.2f *Additional requirement*

- Each doctor works at between one and many outpatient locations, and each outpatient location must have one but can have many doctors working at it.



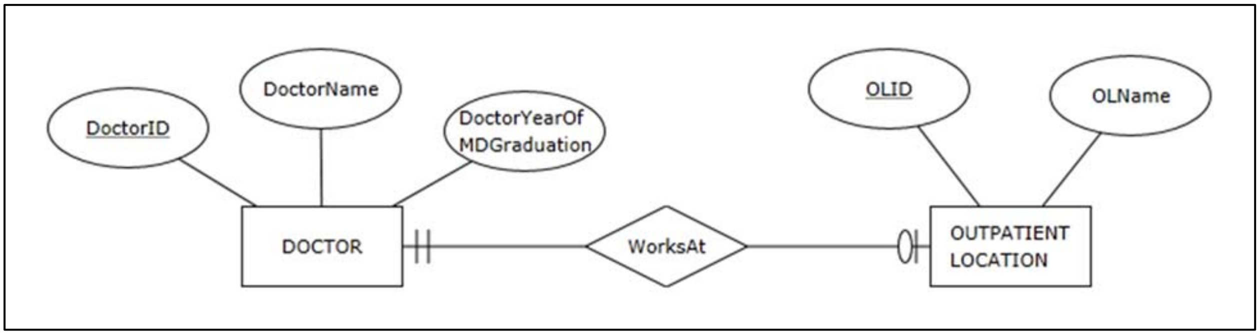
E2.2g *Additional requirement*

- Each doctor works at between none (strictly working in the main hospital) and many outpatient locations, and each outpatient location can have between none (rehab and minor health issues only) and many doctors working at it.



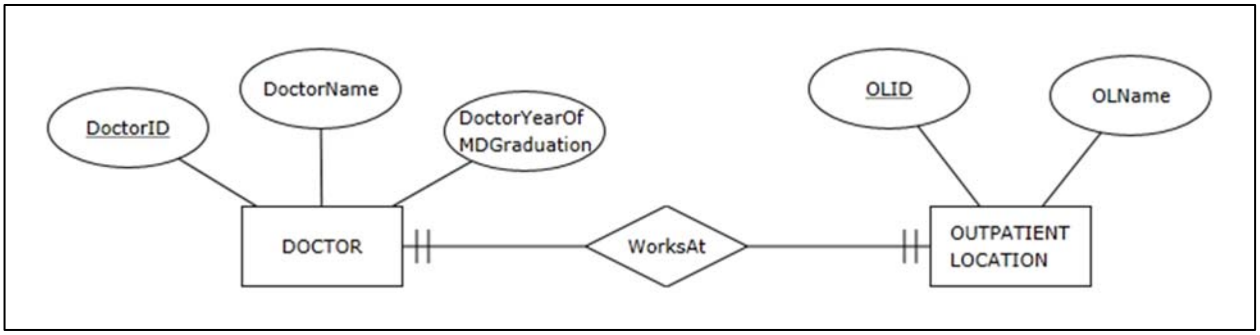
E2.2h *Additional requirement*

- Each doctor works at either one outpatient location or at none (strictly working in the main hospital), and each outpatient location has exactly one doctor working at it.



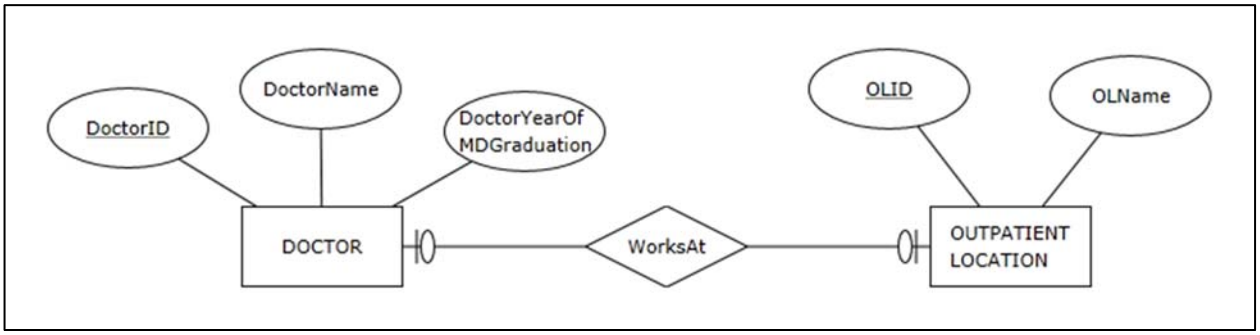
***E2.2i** Additional requirement*

- Each doctor works at exactly one outpatient location, and each outpatient location has exactly one doctor working at it.



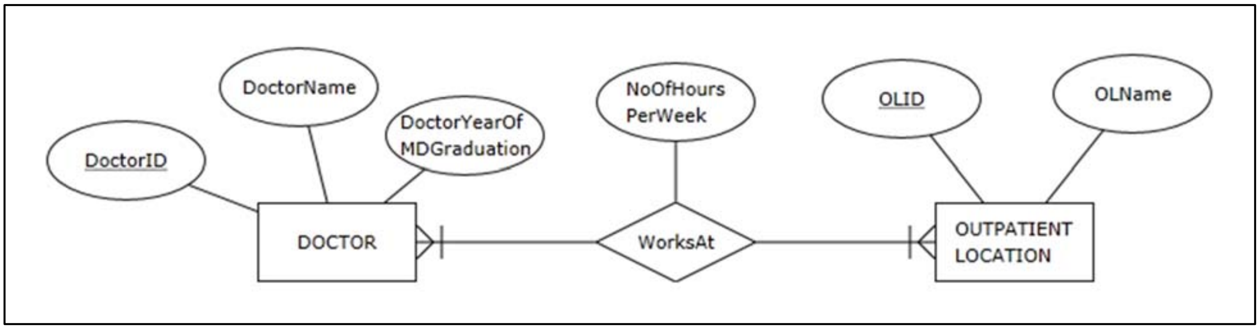
***E2.2j** Additional requirement*

- Each doctor works at either one outpatient location or at none (strictly working in the main hospital), and each outpatient location can have between none (rehab and minor health issues only) and one doctor working at it.



***E2.2k** Additional requirements*

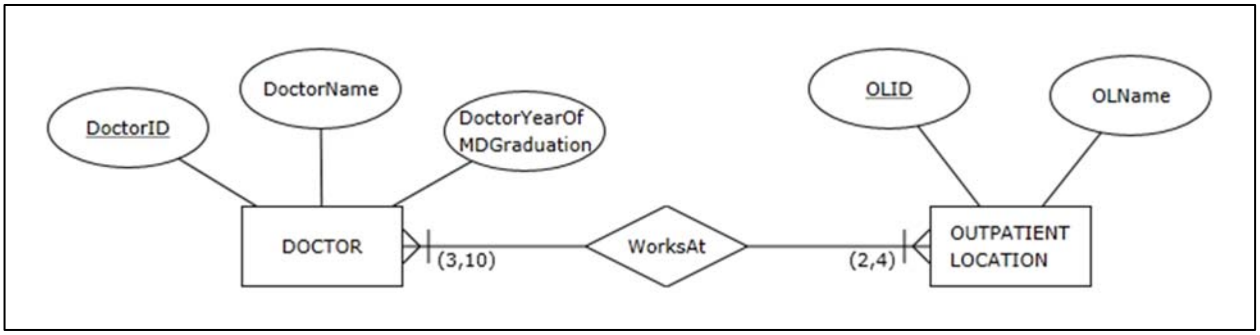
- Each doctor works at between one and many outpatient locations, and each outpatient location must have at least one but can have many doctors working at it.
- For each occurrence of a doctor working at an outpatient location, we keep track of the NoOfHoursPerWeek (number of hours per week that a doctor works at an outpatient location).



Copyright (c) 2020 Nenad Jukic and Prospect Press

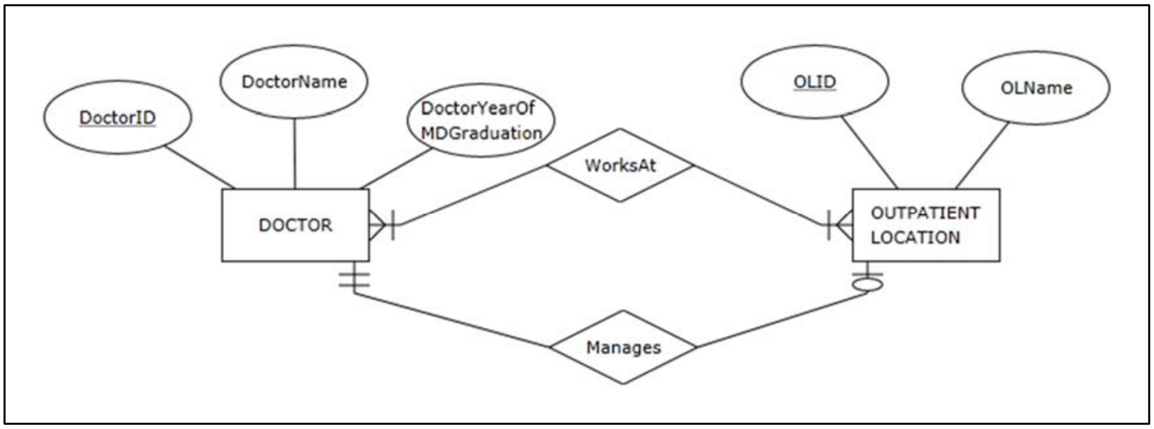
***E2.2l** Additional requirement*

- Each doctor works at between two and four outpatient locations, and each outpatient location has between three and ten doctors working at it.



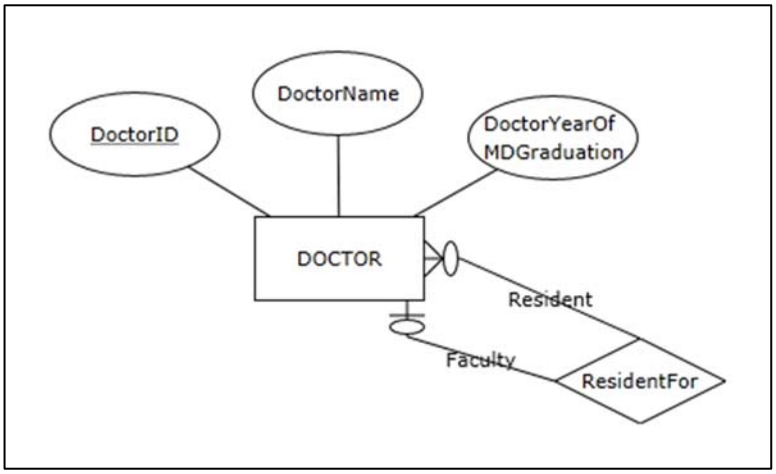
***E2.2m** Additional requirements*

- Each doctor works at between one and many outpatient locations, and each outpatient location must have at least one but can have many doctors working at it.
- Each doctor manages between zero and one outpatient locations, and each outpatient location has exactly one doctor managing it.



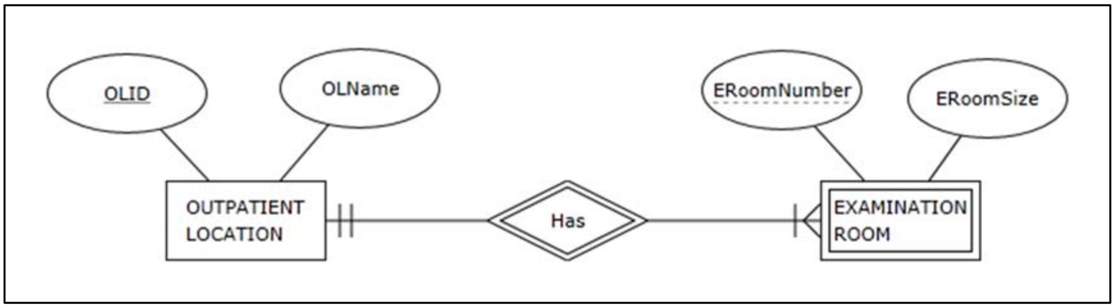
E2.3 *Create the ER diagram based on the following requirements:*

- A health care organization keeps track of its doctors.
- For each doctor it keeps track of the DoctorID (unique), DoctorName, and DoctorYearOfMDGraduation (year of graduating from medical school).
- A doctor can be a resident for either one or none of the faculty doctors. A doctor can have many resident doctors, but does not have to have any.



E2.4 *Create the ER diagram based on the following requirements:*

- A health care organization keeps track of outpatient locations and examination rooms within the outpatient locations.
- For each outpatient location it keeps track of the OLID (unique) and OLName.
- For each examination rooms it keeps track of the ERoomNumber (unique within its outpatient center, but examination rooms in different outpatient locations can have the same ERoomNumber), and ERoomSize.
- Each outpatient location must have at least one but can have many examination rooms. Each examination room is located in one outpatient location.



Any legitimate examples for **E2.5-E2.8** will work.