

# Entity/Relationship Modelling

# Database Design

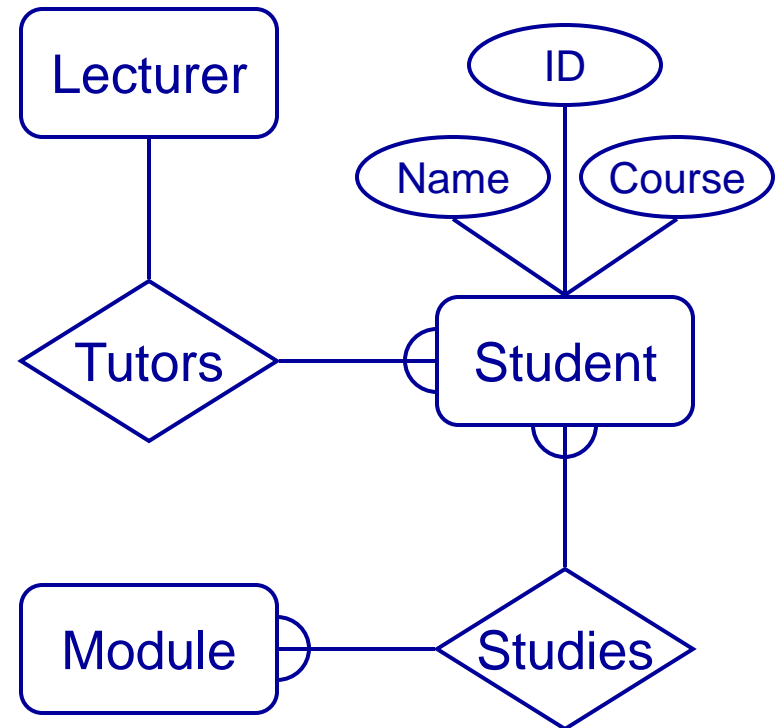
- Conceptual design
  - Build a model independent of the choice of DBMS
- Logical design
  - Create the database in a given DBMS
- Physical design
  - How the database is stored in hardware

# Entity/Relationship Modelling

- E/R Modelling is used for conceptual design
  - Entities - objects or items of interest
  - Attributes - facts about, or properties of, an entity
  - Relationships - links between entities
- Example
  - In a University database we might have entities for Students, Modules and Lecturers. Students might have attributes such as their ID, Name, and Course, and could have relationships with Modules (enrolment)

# Entity/Relationship Diagrams

- E/R Models are often represented as E/R diagrams that
  - Give a conceptual view of the database
  - Are independent of the choice of DBMS
  - Can identify some problems in a design

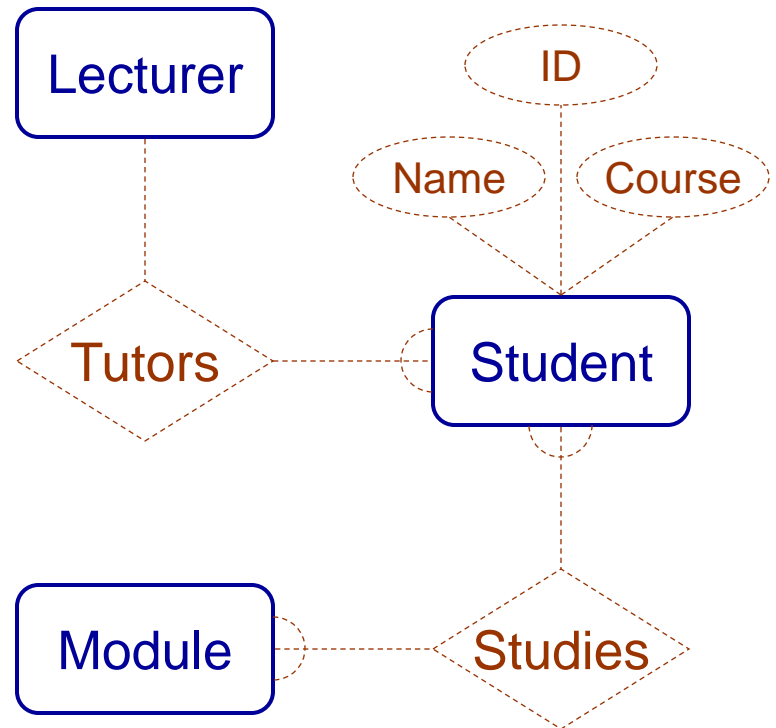


# Entities

- Entities represent objects or things of interest
  - Physical things like students, lecturers, employees, products
  - More abstract things like modules, orders, courses, projects
- Entities have
  - A general type or class, such as Lecturer or Module
  - Instances of that particular type, such as Steve Mills, Natasha Alechina are instances of Lecturer
  - Attributes (such as name, email address)

# Diagramming Entities

- In an E/R Diagram, an entity is usually drawn as a box with rounded corners
- The box is labelled with the name of the class of objects represented by that entity

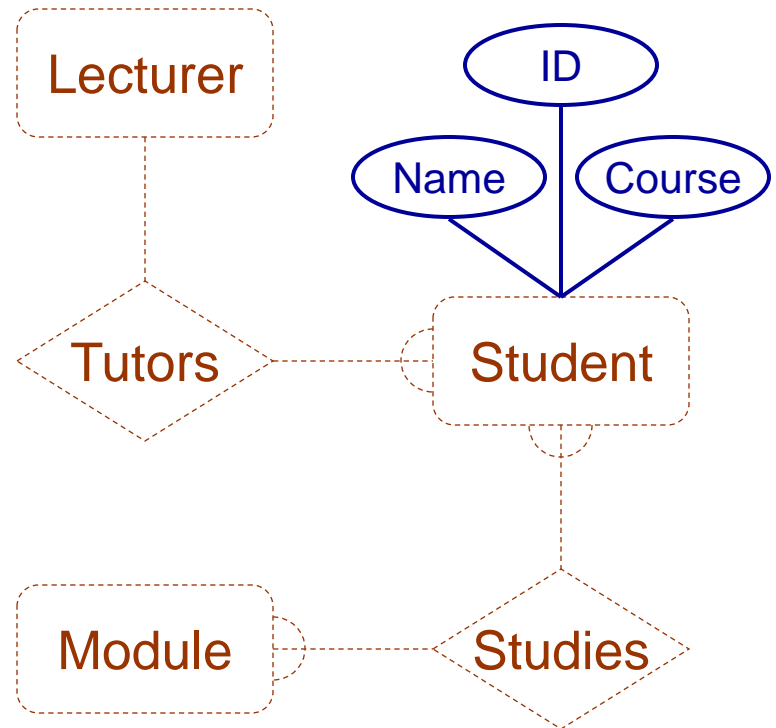


# Attributes

- Attributes are facts, aspects, properties, or details about an entity
  - Students have IDs, names, courses, addresses, ...
  - Modules have codes, titles, credit weights, levels, ...
- Attributes have
  - A name
  - An associated entity
  - Domains of possible values
  - Values from the domain for each instance of the entity they are belong to

# Diagramming Attributes

- In an E/R Diagram attributes may be drawn as ovals
- Each attribute is linked to its entity by a line
- The name of the attribute is written in the oval





# Relationships

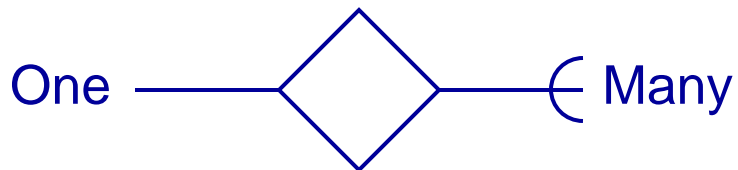
- Relationships are an association between two or more entities
  - Each Student takes several Modules
  - Each Module is taught by a Lecturer
  - Each Employee works for a single Department
- Relationships have
  - A name
  - A set of entities that participate in them
  - A degree - the number of entities that participate (most have degree 2)
  - A cardinality ratio

# Cardinality Ratios

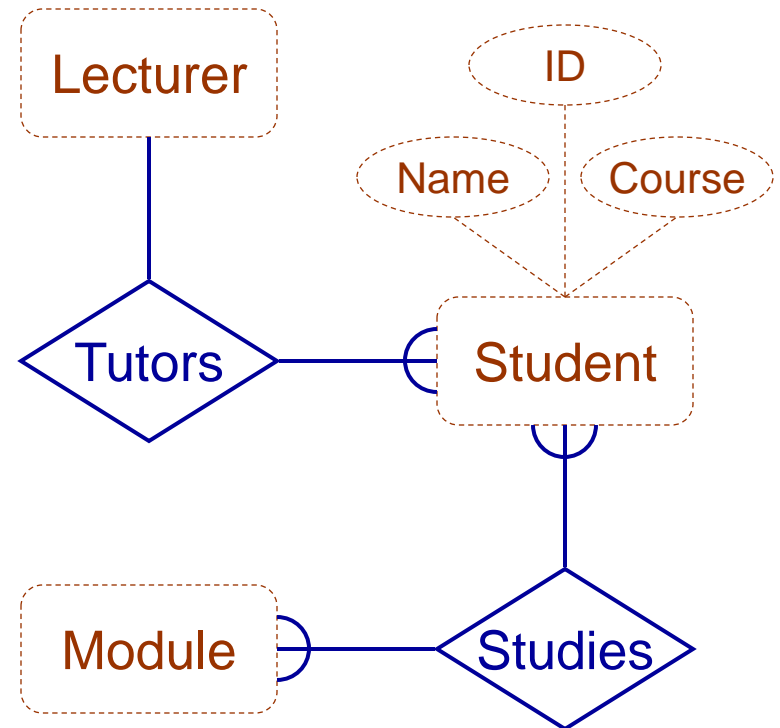
- Each entity in a relationship can participate in zero, one, or more than one instances of that relationship
- This leads to 3 types of relationship...
- One to one (1:1)
  - Each lecturer has a unique office
- One to many (1:M)
  - A lecturer may tutor many students, but each student has just one tutor
- Many to many (M:M)
  - Each student takes several modules, and each module is taken by several students

# Diagramming Relationships

- Relationships are links between two entities
- The name is given in a diamond box
- The ends of the link show cardinality

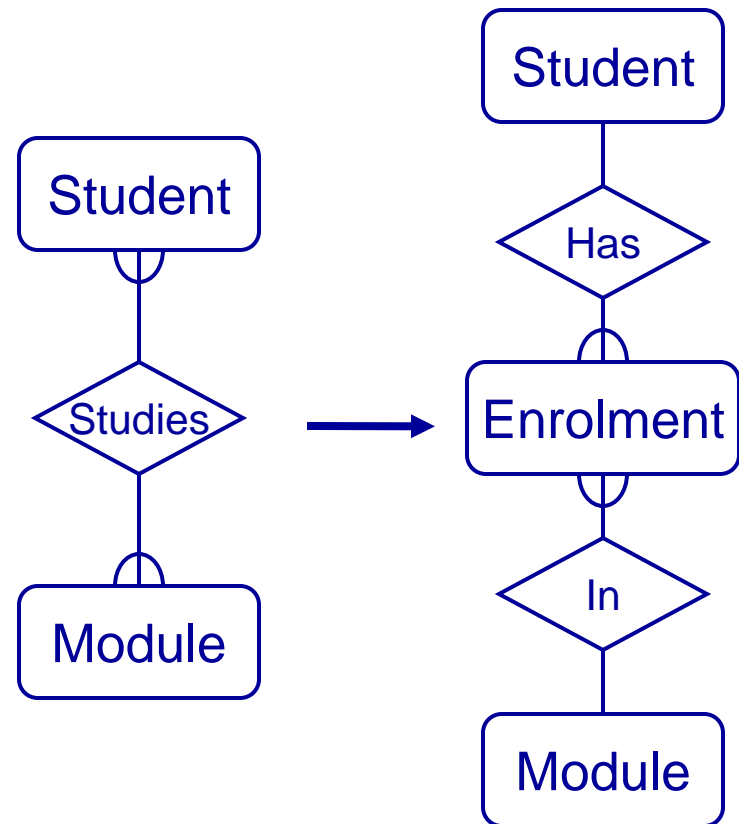


Entity Relationship Modelling



# Removing M:M Relationships

- Many to many relationships are difficult to represent
- We can split a many to many relationship into two one to many relationships
- An entity represents the M:M relationship



# Making E/R Models

- To make an E/R model you need to identify
  - Entities
  - Attributes
  - Relationships
  - Cardinality ratios
- from a description
- General guidelines
  - Since entities are things or objects they are often nouns in the description
  - Attributes are facts or properties, and so are often nouns also
  - Verbs often describe relationships between entities

# Example

A university consists of a number of departments. Each department offers several courses. A number of modules make up each course. Students enrol in a particular course and take modules towards the completion of that course. Each module is taught by a lecturer from the appropriate department, and each lecturer tutors a group of students

# Example - Entities

A university consists of a number of **departments**. Each department offers several **courses**. A number of **modules** make up each course. **Students** enrol in a particular course and take modules towards the completion of that course. Each module is taught by a **lecturer** from the appropriate department, and each lecturer tutors a group of students

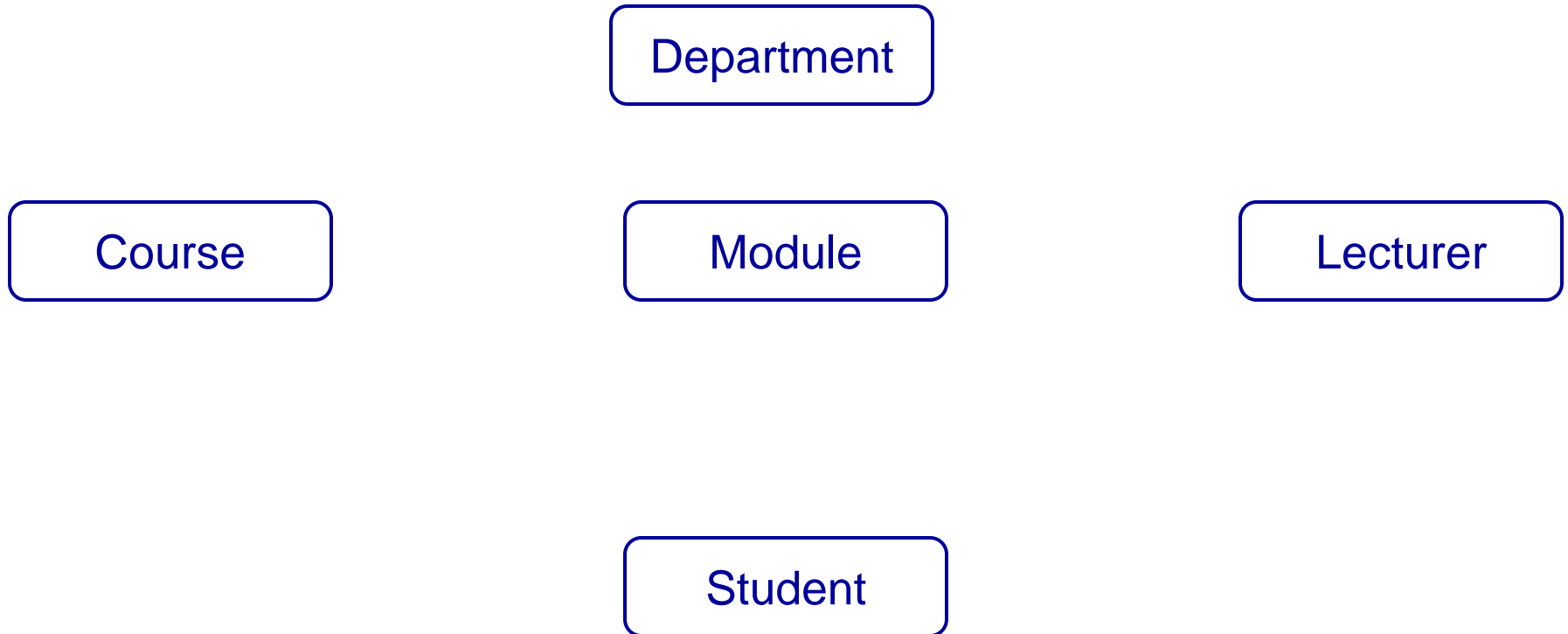
# Example - Relationships

- A university consists of a number of departments. Each department **offers** several courses. A number of modules **make up** each course. Students **enrol in** a particular course and **take** modules towards the completion of that course. Each module is **taught by** a lecturer **from the** appropriate department, and each lecturer **tutors** a group of students



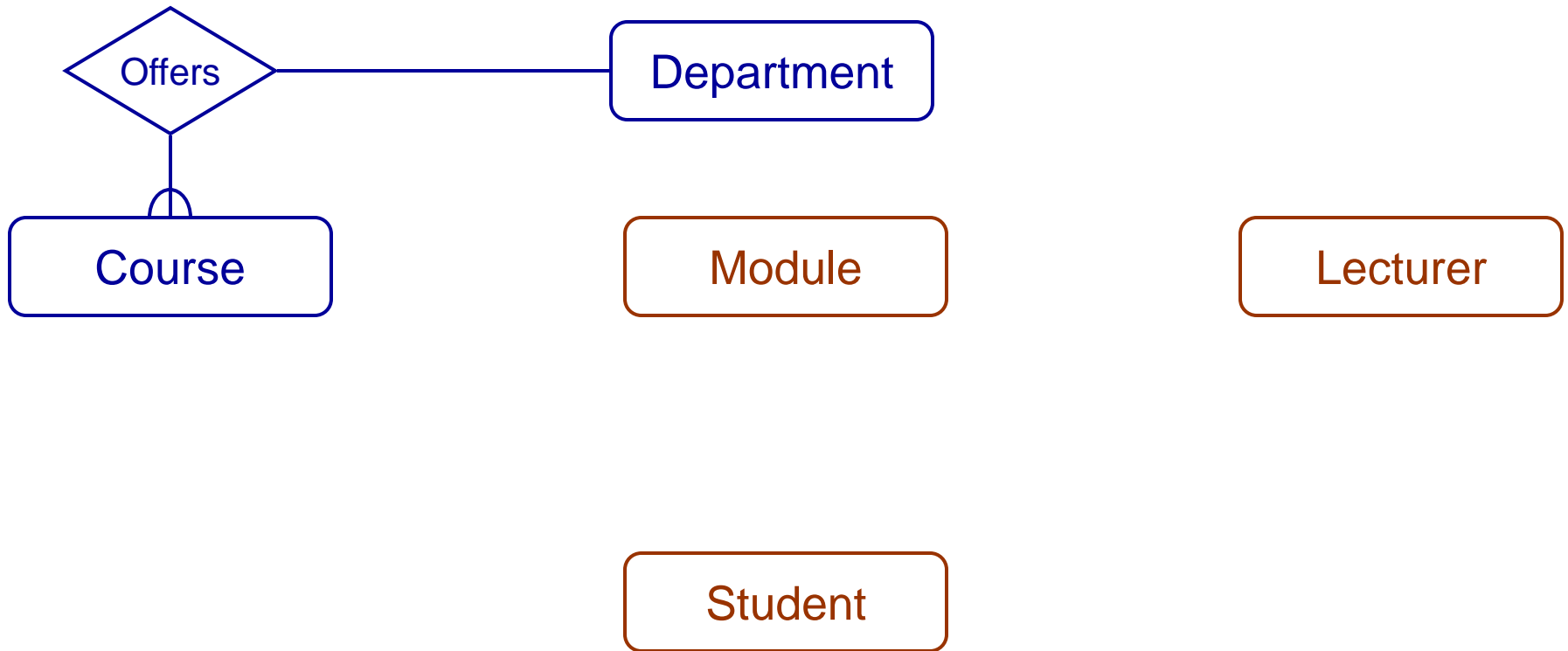
# Example - E/R Diagram

Entities: Department, Course, Module, Lecturer, Student



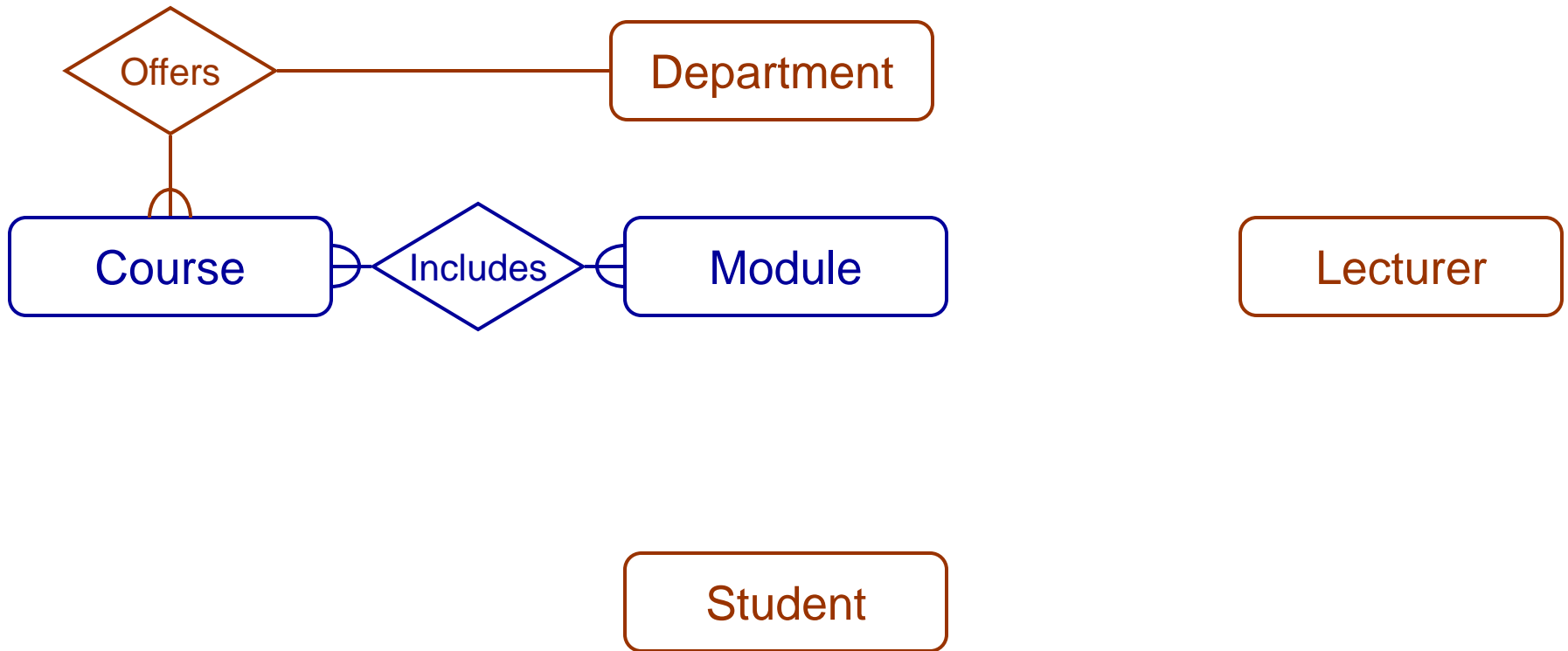
# Example - E/R Diagram

## Each department offers several courses



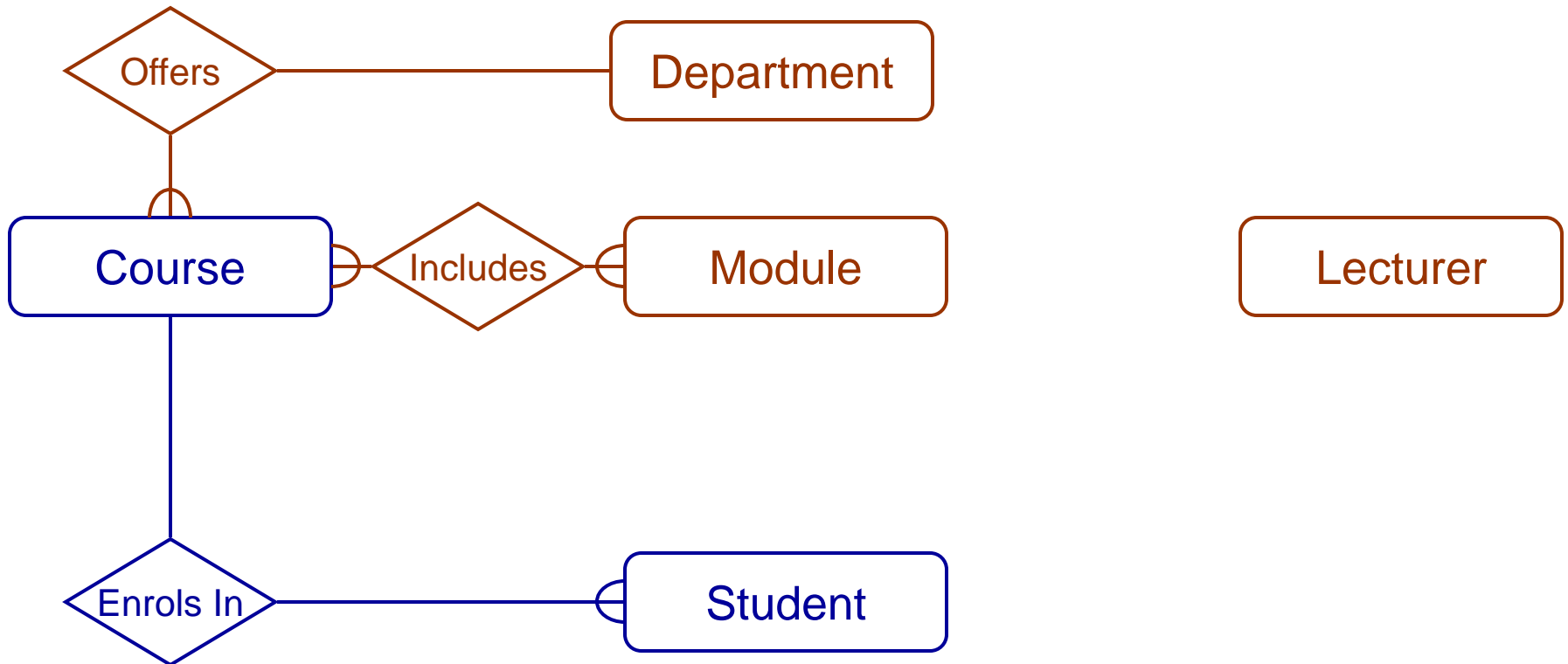
# Example - E/R Diagram

A number of modules **make up** each courses



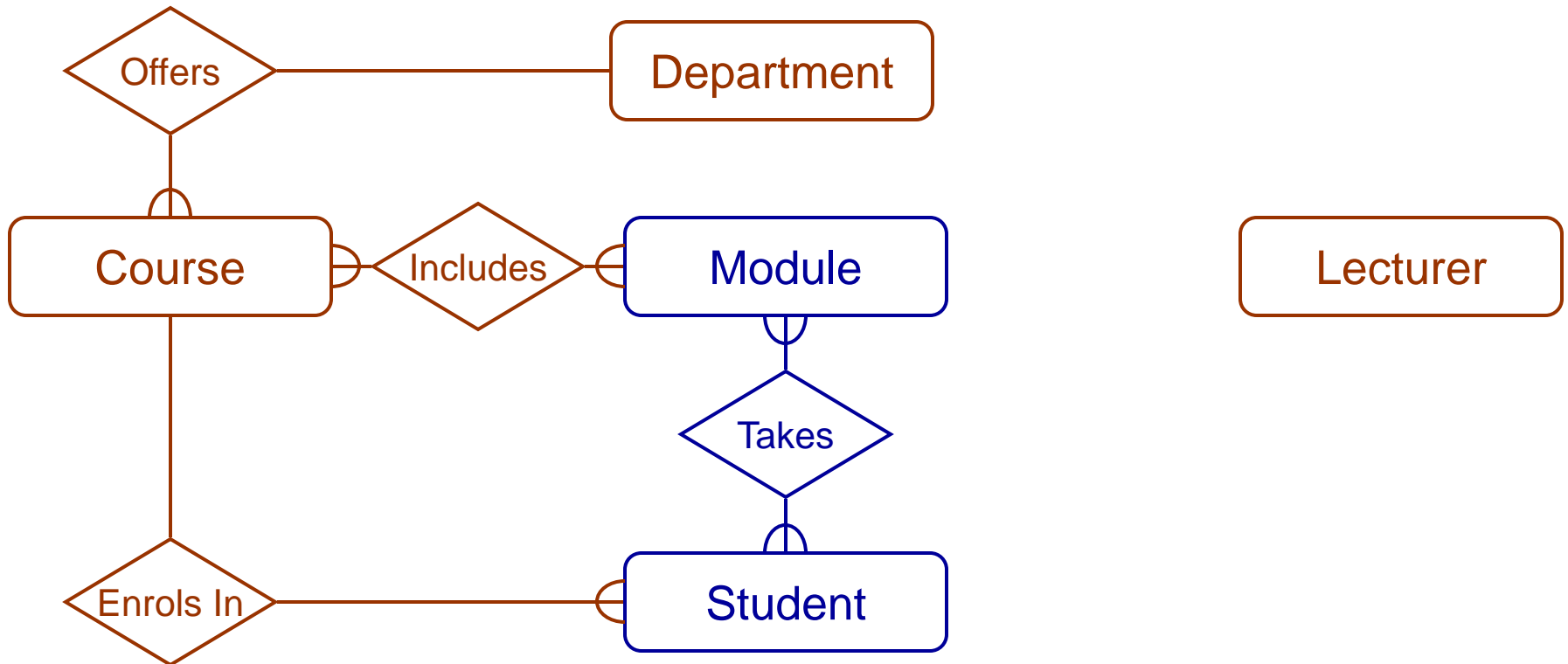
# Example - E/R Diagram

Students **enrol** in a particular course



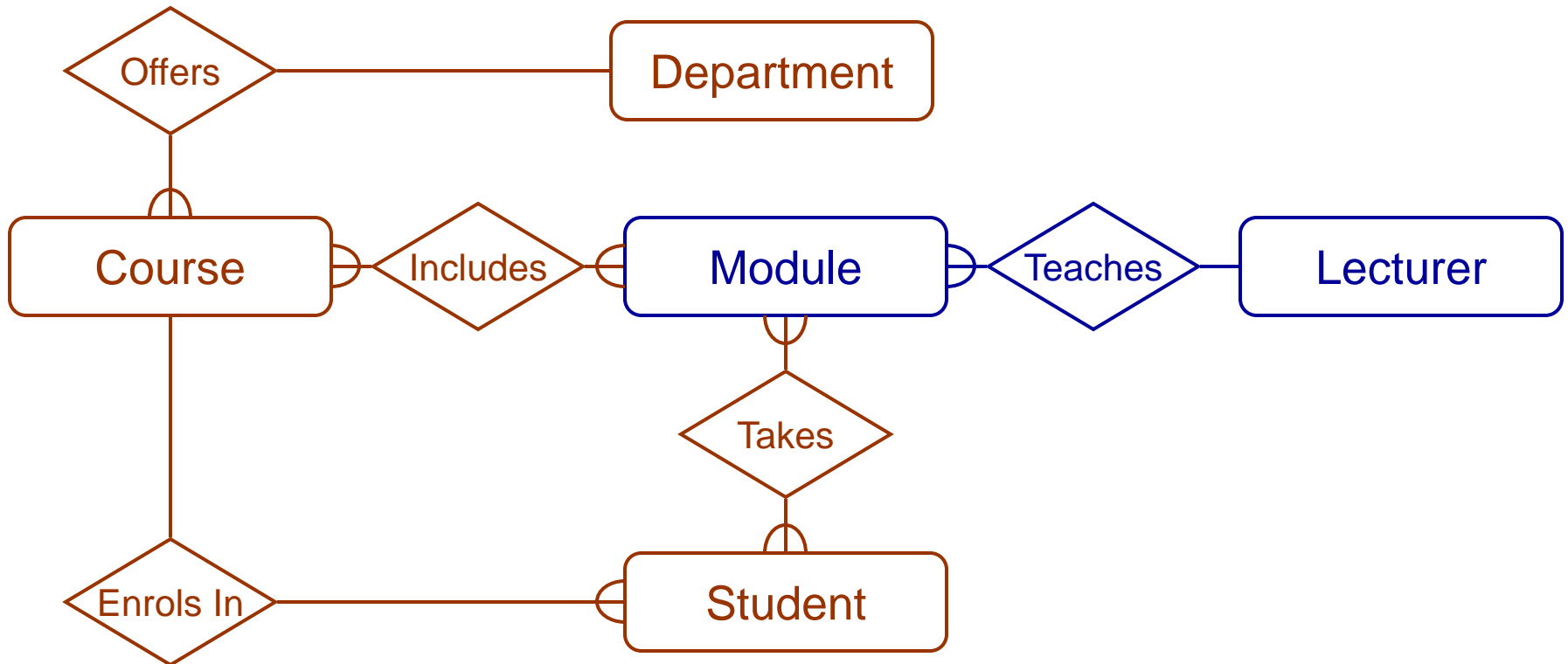
# Example - E/R Diagram

Students ... **take** modules



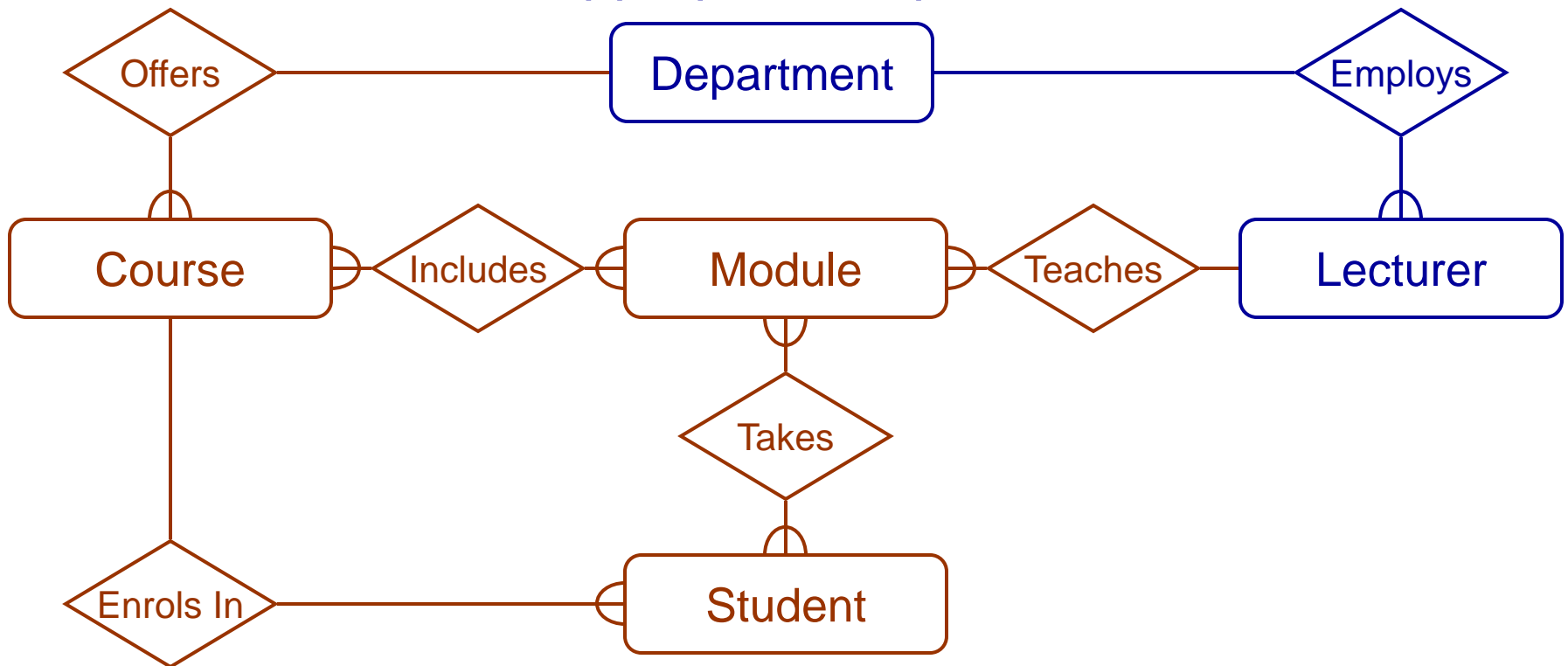
# Example - E/R Diagram

Each module is taught by a lecturer



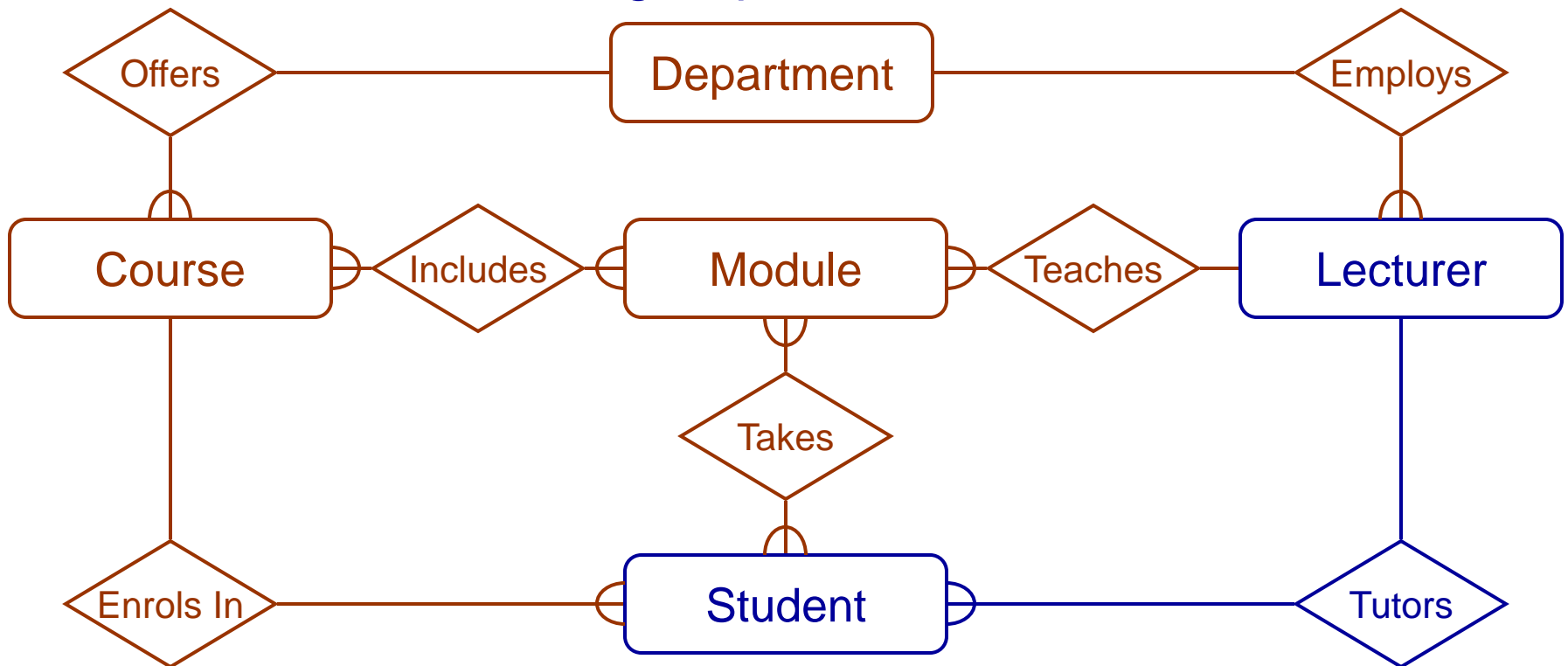
# Example - E/R Diagram

a lecturer from the appropriate department



# Example - E/R Diagram

each lecturer **tutors** a group of students





# Example - E/R Diagram

