

Semester I 2024/2025

Subject : Database (SECD2523) Section : 01 & 09 – Dr Haslina Hashim

Task : LAB 5-1 – ENHANCED ENTITY RELATIONSHIP MODELLING

Instruction:

Students are required to discuss the questions below.

# QUESTION 01 – CAR DEALERSHIP

A car dealership wishes to maintain data about the customers who purchase a car. Each customer may purchase one or more vehicles and each vehicle can be purchased by many different customers over time (for example, a customer may purchase a new vehicle, trade that vehicle in and someone else can purchase the vehicle traded in.) Data that the dealership wishes to keep regarding customers includes customer identification number, name, address, home phone, work phone, cell phone and e- mail address.

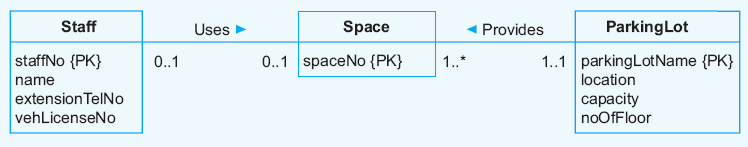
Information about vehicle includes vehicle identification number, make, model, year, transmission type, engine size and color. Vehicles can be a member of one of the following categories: cars, trucks, minivans or SUVS. A vehicle can be a member of only one category at a given time. Trucks, minivans and SUVS have unique attributes – cars do not. Trucks have the following unique attributes: Cab (example: regular, super or crew) and Driver (for example: 94X2 or (4X4); Minivans have Accessory package and SUVS have SUV style.

The dealership is interested in the date of Purchase, amount of sale and Salesperson(s) completing each sale.

# QUESTION 02 – PARKING SPACE

Introduce specialization/generalization concepts into the ER model shown in Figure 1 and to show the following

1. The majority of parking spaces are under cover and each can be allocated for use by a member of staff for a monthly rate.
2. Parking spaces that are not under cover are free to use when available.
3. Up to twenty covered parking spaces are available for use by visitors to the company. However, only members of staff are able to book out a space for the day of the visit. There is no charge for this type of booking, but the member of staff must provide the visitor’s vehicle license number.



**Figure 1 Parking lot ER model**

Description of the ER model above.

* 1. A large organization has several parking lots, which are used by staff.
  2. Each parking lot has a unique name, location, capacity, and number of floors (where appropriate).
  3. Each parking lot has parking spaces, which are uniquely identified using a space number.
  4. Members of staff can request the use of a parking space. Each member of staff has a unique number, name, telephone extension number, and vehicle license number.

# QUESTION 03 – FACULTY OF COMPUTER SCIENCE

At Faculty of Computer Science and Information System, students can be categorized as undergraduate student, postgraduate student, research assistant and teaching assistant. Undergraduate students cannot fall into the other categories; however, postgraduate student can also be either research assistant or teaching assistant. Students register subjects at the faculty that can be categorized into three which are Faculty Compulsory, Department Compulsory, University Compulsory and Elective. Students must take all categories of courses in order to get their degree.

1. What is the multiplicity constraint for Specialization/Generalization? (1 mark )
2. What is the participation and disjoint constraints for the entity Student and Course? (4 marks)
3. Draw the Enhance Entity relationship diagram for the entity Student based on this description with the participation and disjoint constraints. (16 marks)

# QUESTION 04 – UNIVERSITY STUDENTS TRANSCRIPT DATABASE

Consider the following set of requirements for a UNIVERSITY database that is used to keep track of students' transcripts.

* 1. The university keeps track of each student's name, student number, social security number, current address and phone, permanent address and phone, birthdate, sex, class (freshman, sophomore, ..., graduate), major department, minor department (if any), and degree program (B.A., B.S., ..., Ph.D.). Some user applications need to refer to the city, state, and zip of the student's permanent address, and to the student's last name. Both social security number and student number have unique values for each student.
  2. Each department is described by a name, department code, office number, office phone, and college. Both name and code have unique values for each department.
  3. Each course has a course name, description, course number, number of semester hours, level, and offering department. The value of the course number is unique for each course.
  4. Each section has an instructor, semester, year, course, and section number. The section number distinguishes different sections of the same course that are taught during the same semester/year; its values are 1, 2, 3, ..., up to the number of sections taught during each semester.
  5. A grade report has a student, section, letter grade, and numeric grade (0, 1, 2, 3, 4 for F, D, C, B, A, respectively).

Design an ER schema for this application and draw an ER diagram for that schema. Specify key attributes of each entity type and structural constraints on each relationship type. Note any unspecified requirements and make appropriate assumptions to make the specification complete.

### **QUESTION 01 – CAR DEALERSHIP**

#### Requirements:

* A car dealership wants to maintain data about customers and vehicles.
* Vehicles can be purchased by multiple customers over time.
* Vehicle categories: Cars, Trucks, Minivans, SUVs.
* Unique attributes:
  + Trucks: Cab and Driver type.
  + Minivans: Accessory package.
  + SUVs: SUV style.
* Sales information: Date of purchase, sale amount, and salesperson(s).

#### Enhanced ER Diagram:

An EER diagram will be created with the following details:

1. **Entities**:
   * Customer: Attributes: CustomerID (PK), Name, Address, HomePhone, WorkPhone, CellPhone, Email.
   * Vehicle: Attributes: VehicleID (PK), Make, Model, Year, TransmissionType, EngineSize, Color.
   * Sale: Attributes: SaleID (PK), PurchaseDate, SaleAmount.
   * Specialized entities: Truck, Minivan, SUV with unique attributes.
2. **Relationships**:
   * Purchase: Between Customer and Vehicle (many-to-many).
   * SoldBy: Between Sale and Salesperson.

@startuml

entity Customer {

\* CustomerID : PK

--

Name

Address

HomePhone

WorkPhone

CellPhone

Email

}

entity Vehicle {

\* VehicleID : PK

--

Make

Model

Year

TransmissionType

EngineSize

Color

}

entity Sale {

\* SaleID : PK

--

PurchaseDate

SaleAmount

}

entity Truck {

+ CabType

+ DriverType

}

entity Minivan {

+ AccessoryPackage

}

entity SUV {

+ SUVStyle

}

entity Salesperson {

\* SalespersonID : PK

--

Name

}

Customer "1" -- "0..\*" Sale

Sale "0..\*" -- "1" Vehicle

Vehicle <|-- Truck

Vehicle <|-- Minivan

Vehicle <|-- SUV

Sale "0..\*" -- "1..\*" Salesperson

@enduml

### **QUESTION 02 – PARKING SPACE**

#### Specialization/Generalization for Parking Space:

1. **Entities**:
   * ParkingSpace: Attributes: SpaceNumber (PK), Type.
   * Specialized entities:
     + CoveredParking: Attributes: MonthlyRate.
     + VisitorParking: Attributes: VisitorLicensePlate.
     + UncoveredParking.
2. **Relationships**:
   * AllocatedTo: Between CoveredParking and Staff.
   * BookedBy: Between VisitorParking and Staff.

#### Steps:

* Covered parking can have a MonthlyRate and is allocated to staff.
* Visitor parking must track the license number and be booked by staff.
* Uncovered parking is available for free and does not require additional tracking.

@startuml

entity ParkingSpace {

\* SpaceNumber : PK

--

Type

}

entity CoveredParking {

+ MonthlyRate

}

entity VisitorParking {

+ VisitorLicensePlate

}

entity UncoveredParking

entity Staff {

\* StaffID : PK

--

Name

Telephone

VehicleLicense

}

ParkingSpace <|-- CoveredParking

ParkingSpace <|-- VisitorParking

ParkingSpace <|-- UncoveredParking

CoveredParking "0..\*" -- "1..\*" Staff : AllocatedTo

VisitorParking "0..\*" -- "1..\*" Staff : BookedBy

@enduml

### **QUESTION 03 – FACULTY OF COMPUTER SCIENCE**

#### Requirements:

* Students can be categorized as:
  + Undergraduate.
  + Postgraduate.
  + ResearchAssistant.
  + TeachingAssistant.
* Courses have categories: FacultyCompulsory, DepartmentCompulsory, UniversityCompulsory, Elective.

#### Answers:

**(a) Multiplicity Constraint**: For specialization/generalization, a student can belong to at most one category (Undergraduate) or multiple categories (Postgraduate, ResearchAssistant, TeachingAssistant).

**(b) Participation and Disjoint Constraints**:

* Participation: Total for Student in all categories.
* Disjoint: Undergraduate is disjoint from others, while Postgraduate may overlap with ResearchAssistant and TeachingAssistant.

**(c) Enhanced ER Diagram**: Entities:

1. Student: StudentID (PK), Name, Age.
   * Specializations: Undergraduate, Postgraduate with subclasses ResearchAssistant, TeachingAssistant.
2. Course: Attributes: CourseID (PK), Category.
3. Relationships:
   * Registers: Between Student and Course.

@startuml

entity Student {

\* StudentID : PK

--

Name

Age

}

entity Course {

\* CourseID : PK

--

Category

}

entity Undergraduate

entity Postgraduate

entity ResearchAssistant

entity TeachingAssistant

Student <|-- Undergraduate

Student <|-- Postgraduate

Postgraduate <|-- ResearchAssistant

Postgraduate <|-- TeachingAssistant

Student "1..\*" -- "0..\*" Course : Registers

@enduml

### **QUESTION 04 – UNIVERSITY STUDENTS TRANSCRIPT DATABASE**

#### Requirements:

1. **Entities**:
   * Student: Attributes: StudentID (PK), Name, SSN, CurrentAddress, PermanentAddress, BirthDate, Sex, Class, MajorDept, MinorDept, DegreeProgram.
   * Department: Attributes: DepartmentCode (PK), Name, OfficeNumber, OfficePhone, College.
   * Course: Attributes: CourseNumber (PK), CourseName, Description, SemesterHours, Level, OfferingDept.
   * Section: Attributes: SectionNumber (PK), Semester, Year, Instructor.
   * GradeReport: Attributes: LetterGrade, NumericGrade.
2. **Relationships**:
   * EnrollsIn: Between Student and Section.
   * Offers: Between Department and Course.
   * Teaches: Between Instructor and Section.

#### ER Diagram:

* Key attributes identified for all entities.
* Relationship constraints:
  + Many-to-many for Student and Section.
  + One-to-many for Department to Course.

@startuml

entity Student {

\* StudentID : PK

--

Name

SSN

CurrentAddress

PermanentAddress

BirthDate

Sex

Class

MajorDept

MinorDept

DegreeProgram

}

entity Department {

\* DepartmentCode : PK

--

Name

OfficeNumber

OfficePhone

College

}

entity Course {

\* CourseNumber : PK

--

CourseName

Description

SemesterHours

Level

OfferingDept

}

entity Section {

\* SectionNumber : PK

--

Semester

Year

Instructor

}

entity GradeReport {

+ LetterGrade

+ NumericGrade

}

Student "1..\*" -- "0..\*" Section : EnrollsIn

Section "1..\*" -- "1" Course

Course "0..\*" -- "1" Department : Offers

Section "0..\*" -- "1" Instructor : Teaches

GradeReport "1" -- "1" Section

GradeReport "1" -- "1" Student

@enduml