

**Semester I 2021/2022**

Subject : Database (SECD2523)

Section : 09 – Dr Haslina Hashim

Task : LAB 4-1 – ENTITY RELATIONSHIP MODELLING

## Instruction:

Students are required to discuss the questions below.

## **QUESTION 01 – USA TYPE GEOGRAPHY**

|  |
| --- |
| Model the USA-type geography, where the following is of interest:   * States of a country including their name, population, area, and date established. * Cities including their name, population, date established and founder's name. * Rivers including their name and length. * The capital city of each state, including the date declared. * The source of each river, i.e., the state in which it begins (which may be none) * The fact that cities may be located on particular rivers. * The fact that cities are in states. * The length of each river within each state. * Information about states that adjoin each another, and the length of their common border. * Assume that the names of states and rivers are unique. City names are only unique within a state. |

Изображение выглядит как текст, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

Table State {

StateID int [pk]

Name varchar [unique]

Population int

Area float

DateEstablished date

}

Table City {

CityID int [pk]

Name varchar

Population int

DateEstablished date

FounderName varchar

StateID int [ref: > State.StateID]

}

Table River {

RiverID int [pk]

Name varchar [unique]

Length float

SourceStateID int [ref: > State.StateID, null]

}

Table CapitalCity {

StateID int [pk, ref: > State.StateID]

CityID int [ref: > City.CityID]

DateDeclared date

}

Table CityRiver {

CityID int [pk, ref: > City.CityID]

RiverID int [pk, ref: > River.RiverID]

}

Table RiverState {

RiverID int [pk, ref: > River.RiverID]

StateID int [pk, ref: > State.StateID]

LengthInState float

}

Table StateBorder {

State1ID int [pk, ref: > State.StateID]

State2ID int [pk, ref: > State.StateID]

BorderLength float

}

## **QUESTION 02 – VIDEO RENTAL COMPANY**

Read the following case study, which describes the data requirements for a video rental company. The video rental company has several branches throughout the USA. The data held on each branch is the branch address made up of street, city, state, and zip code, and the telephone number. Each branch is given a branch number, which is unique throughout the company. Each branch is allocated staff, which includes a Manager. The Manager is responsible for the day-to-day running of a given branch. The data held on a member of staff is his or her name, position, and salary. Each member of staff is given a staff number, which is unique throughout the company. Each branch has a stock of videos. The data held on a video is the catalogue number, video number, title, category, daily rental, cost, status, and the names of the main actors, and the director. The catalogue number uniquely identifies each video. However, in most cases, there are several copies of each video at a branch, and the individual copies are identified using the video number. A video is given a category such as Action, Adult, Children, Drama, Horror, or Sci-Fi. The status indicates whether a specific copy of a video is available for rent. Before hiring a video from the company, a customer must first register as a member of a local branch. The data held on a member is the first and last name, address, and the date that the member registered at a branch. Each member is given a member number, which is unique throughout all branches of the company. Once registered, a member is free to rent videos, up to maximum of ten at any one time. The data held on each video rented is the rental number, the full name and number of the member, the video number, title, and daily rental, and the dates the video is rented out and date returned. The rental number is unique throughout the company.

1. Identify the main entity types of the video rental company.
2. Identify the main relationship types between the entity types described in (a) and represent each relationship as an ER diagram.
3. Determine the multiplicity constraints for each relationship described in (b). Represent the multiplicity for each relationship in the ER diagrams created in (b).
4. Identify attributes and associate them with entity or relationship types. Represent each attribute in the ER diagrams created in (c).
5. Determine candidate and primary key attributes for each (strong) entity type.
6. Using your answers (a) to (e) attempt to represent the data requirements of the video rental company as a single ER diagram. State any assumptions necessary to support your design.

Изображение выглядит как снимок экрана, текст

Контент, сгенерированный ИИ, может содержать ошибки.

Table Branch {

BranchID int [pk]

Address varchar

TelephoneNumber varchar

}

Table Staff {

StaffID int [pk]

Name varchar

Position varchar

Salary decimal

BranchID int [ref: > Branch.BranchID]

}

Table Video {

CatalogueNumber int [pk]

VideoNumber int

Title varchar

CategoryID int [ref: > Category.CategoryID]

DailyRental decimal

Cost decimal

Status varchar

MainActors varchar

Director varchar

BranchID int [ref: > Branch.BranchID]

}

Table Category {

CategoryID int [pk]

CategoryName varchar

}

Table Member {

MemberID int [pk]

FirstName varchar

LastName varchar

Address varchar

DateRegistered date

BranchID int [ref: > Branch.BranchID]

}

Table Rental {

RentalID int [pk]

MemberID int [ref: > Member.MemberID]

VideoNumber int

Title varchar

DailyRental decimal

RentalDate date

ReturnDate date

}

Ref: Branch.BranchID > Staff.BranchID {1..N}

Ref: Branch.BranchID > Video.BranchID {1..N}

Ref: Branch.BranchID > Member.BranchID {1..N}

Ref: Category.CategoryID > Video.CategoryID {1..N}

Ref: Member.MemberID > Rental.MemberID {1..N}

Ref: Video.VideoNumber > Rental.VideoNumber {1..N}

## **QUESTION 03 – IT TRAINING COMPANY**

You are required to create a conceptual data model of the data requirements for a company that specializes in IT training. The Company has 30 instructors and can handle up to 100 trainees per training session. The Company offers five advanced technology courses, each of which is taught by a teaching team of two or more instructors. Each instructor is assigned to a maximum of two teaching teams or may be assigned to do research. Each trainee undertakes one advanced technology course per training session.

1. Identify the main entity types for the company.
2. Identify the main relationship types and specify the multiplicity for each relationship. State any assumptions you make about the data.
3. Using your answers for (a) and (b), draw a single ER diagram to represent the data requirements for the company.

Изображение выглядит как текст, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

Table Instructor {

InstructorID int [pk]

Name varchar

Email varchar

Phone varchar

Research boolean

}

Table Trainee {

TraineeID int [pk]

Name varchar

Email varchar

Phone varchar

}

Table Course {

CourseID int [pk]

CourseName varchar

Description text

}

Table TeachingTeam {

TeamID int [pk]

CourseID int [ref: > Course.CourseID]

}

Table TrainingSession {

SessionID int [pk]

CourseID int [ref: > Course.CourseID]

StartDate date

EndDate date

}

Ref: Instructor < TeachingTeam.TeamID {0..2}

Ref: TeachingTeam.CourseID > Course.CourseID {1..1}

Ref: TrainingSession.CourseID > Course.CourseID {1..1}

Ref: Trainee.SessionID > TrainingSession.SessionID {1..100}

## **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.

Table Student {

StudentID int [pk]

SSN varchar [unique]

FirstName varchar

LastName varchar

BirthDate date

Sex varchar

Class varchar

MajorDepartment int [ref: > Department.DepartmentID]

MinorDepartment int [ref: > Department.DepartmentID, null]

DegreeProgram varchar

CurrentAddress varchar

CurrentPhone varchar

PermanentAddress varchar

PermanentPhone varchar

PermanentCity varchar

PermanentState varchar

PermanentZip varchar

}

Table Department {

DepartmentID int [pk]

DepartmentName varchar [unique]

OfficeNumber varchar

OfficePhone varchar

College varchar

}

Table Course {

CourseID int [pk]

CourseName varchar

Description text

CourseNumber varchar [unique]

SemesterHours int

Level varchar

OfferingDepartment int [ref: > Department.DepartmentID]

}

Table Section {

SectionID int [pk]

CourseID int [ref: > Course.CourseID]

Instructor varchar

Semester varchar

Year int

SectionNumber int

}

Table GradeReport {

GradeReportID int [pk]

StudentID int [ref: > Student.StudentID]

SectionID int [ref: > Section.SectionID]

LetterGrade char

NumericGrade int

}

