

Name: Vaibhav Patel (Email: vaipate@siue.edu)

Course: CS-434

Instructor: Dr. Mark McKenney

Project 2

- **Part 2 - Translating E/R diagram to Relations:**

Here is the relational schema using the notation of Section 2.2.7. Also, I have listed functional dependencies for each relation. Primary keys are underlined. All constraints are listed, if there are any. Stated Data Types for each attribute.

1. InsurancePlan (

<u>PlanNo</u>	INT,
PlanName	CHAR(100),
PlanType	CHAR(100),
Benefits	VARCHAR(1000),
Rate	INT,
Deductible	INT

)

PlanNo -> PlanName, PlanType, Benefits, Rate, Deductible

2. Customer (

<u>CID</u>	INT,
Name	CHAR(30),
DOB	DATE,
Phone	INT,

```

        SSN            INT,

        CustomerStreet  VARCHAR(255),

        CustomerZipCode INT

    )

```

CID -> Name, DOB, Phone, SSN, CustomerStreet, CustomerZipCode

We can set CID as auto increment number so when inserting new customer in database we do not have to worry about manually entering it.

### 3. Address (

```

        Street         VARCHAR(255),

        ZipCode        INT,

        City           CHAR(30),

        State          CHAR(30)

    )

```

Street, ZipCode -> City, State

### 4. Accident (

```

        ReportNo       INT,

        Date           DATE,

        Description     VARCHAR(1000),

        AccidentStreet  VARCHAR(255),

        AccidentZip     INT

    )

```

ReportNo -> Date, Description, AccidentStreet, AccidentZip

We can set ReportNo as auto increment number so when inserting new accident report in database we do not have to worry about manually entering it.

5. HouseInfo (

<u>HID</u>	INT,
BuildYear	INT,
Type	CHAR(30),
Price	INT,
PlanNo	INT

)

HID -> BuildYear, Type, Price, PlanNo

Here, we can set constraint to our database for BuildYear in case someone enter 3-digit number or value that does not make sense.

6. CarInfo (

<u>VIN</u>	CHAR(30),
Make	CHAR(30),
Model	CHAR(30),
Year	INT,
Color	CHAR(30),
LicensePlate	CHAR(15),
PlanNo	INT

)

VIN -> Make, Model, Year, Color, LicensePlate, PlanNo

Here, we can set constraint for Year attribute, so the value makes sense.

7. Agent (

<u>AID</u>	INT,
Name	CHAR(30),

Phone INT

)

AID -> Name, Phone

8. HousePolicy (

PlanNo INT

)

9. CarPolicy (

PlanNo INT

)

10. Contract (

PlanNo INT,

CID INT,

AID INT

)

PlanNo -> CID, AID

11. CarInvolved (

VIN CHAR(30),

ReportNo INT,

RepairCost INT

)

VIN, ReportNo -> RepairCost

12. CustomerInvolved (

CID INT,

ReportNo INT,

IsGuilty            BOOL

)

CID, ReportNo -> IsGuilty

- **Part 3 – Q&A:**

There are no flaws in the relational database schema in part 2. There is no opportunity to combine relations without introducing redundancy because the way relations are created all tables are Normalized in BCNF forms. For Address table, I thought about combining with Customer table but I am also using address for accidents so why not make separate table which can have relationships with both tables. All the keys on the left side functionally determines right side of FD's. All relations fully satisfy BCNF form so we do not need further Normalization. I have named all attributes appropriately so they make sense in real world.