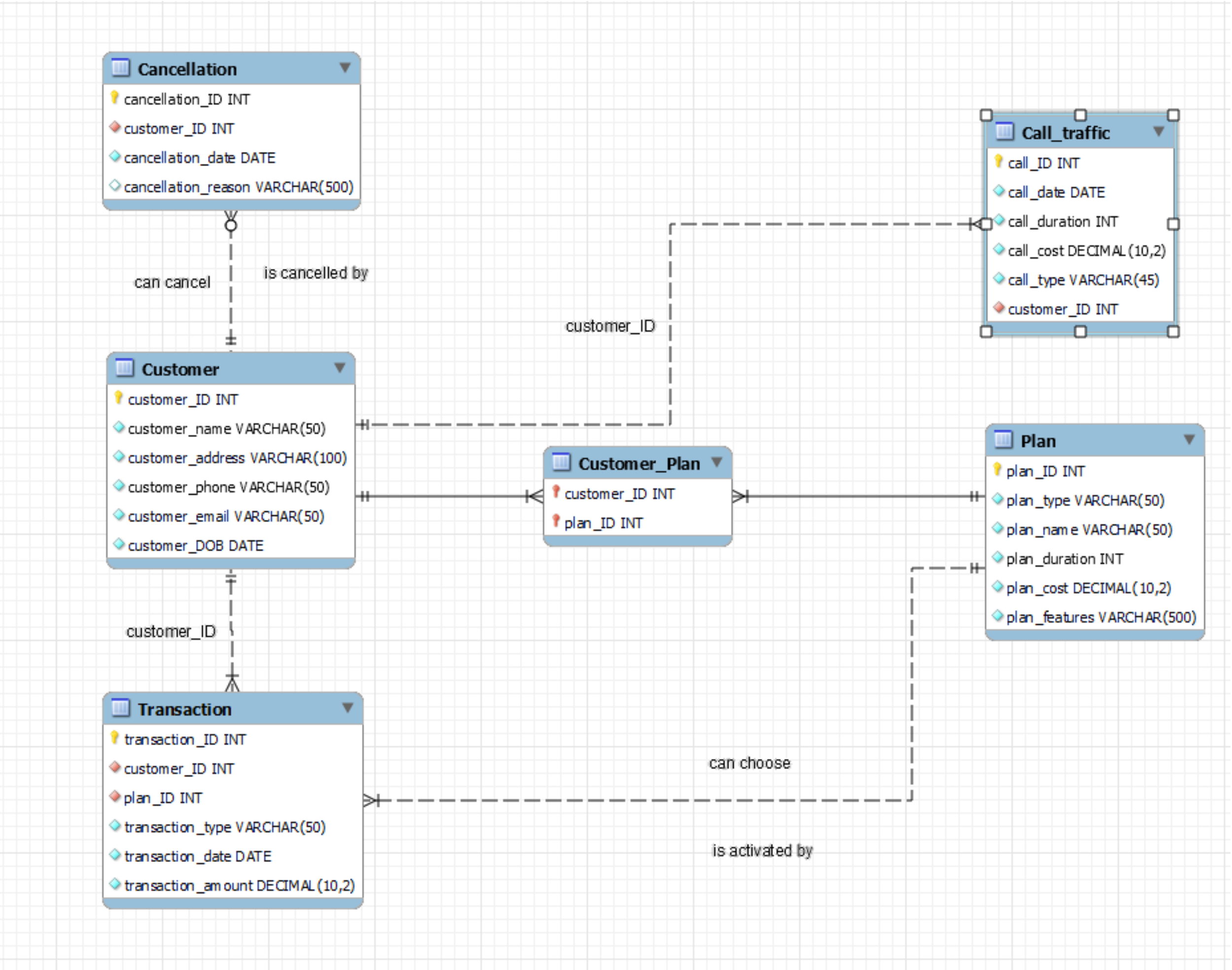
CSCI3287 Homework7

Chengming Li

1. Database Design



2. Create the database

```
-- MySQL Script generated by MySQL Workbench
-- Wed Apr 26 20:33:01 2023
-- Model: New Model    Version: 1.0
-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN
_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUB
STITUTION';

-- -----------------------------------------------------
-- Schema mydb
-- -----------------------------------------------------

-- -----------------------------------------------------
-- Schema mydb
-- -----------------------------------------------------
CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;
USE `mydb` ;

-- -----------------------------------------------------
-- Table `mydb`.`Customer`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `mydb`.`Customer` ;

CREATE TABLE IF NOT EXISTS `mydb`.`Customer` (
  `customer_ID` INT NOT NULL,
  `customer_name` VARCHAR(50) NOT NULL,
  `customer_address` VARCHAR(100) NOT NULL,
  `customer_phone` VARCHAR(50) NOT NULL,
  `customer_email` VARCHAR(50) NOT NULL,
  `customer_DOB` DATE NOT NULL,
  PRIMARY KEY (`customer_ID`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Plan`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `mydb`.`Plan` ;

CREATE TABLE IF NOT EXISTS `mydb`.`Plan` (
  `plan_ID` INT NOT NULL,
  `plan_type` VARCHAR(50) NOT NULL,
  `plan_name` VARCHAR(50) NOT NULL,
  `plan_duration` INT NOT NULL,
  `plan_cost` DECIMAL(10,2) NOT NULL,
  `plan_features` VARCHAR(500) NOT NULL,
  PRIMARY KEY (`plan_ID`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Transaction`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `mydb`.`Transaction` ;

CREATE TABLE IF NOT EXISTS `mydb`.`Transaction` (
  `transaction_ID` INT NOT NULL,
  `customer_ID` INT NOT NULL,
  `plan_ID` INT NOT NULL,
  `transaction_type` VARCHAR(50) NOT NULL,
  `transaction_date` DATE NOT NULL,
  `transaction_amount` DECIMAL(10,2) NOT NULL,
  PRIMARY KEY (`transaction_ID`),
  INDEX `customer_ID_idx` (`customer_ID` ASC) VISIBLE,
  INDEX `plan_ID_idx` (`plan_ID` ASC) VISIBLE,
  CONSTRAINT `customer_planfk_ID`
    FOREIGN KEY (`customer_ID`)
    REFERENCES `mydb`.`Customer` (`customer_ID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `plan_planfk_ID`
    FOREIGN KEY (`plan_ID`)
    REFERENCES `mydb`.`Plan` (`plan_ID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Call_traffic`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `mydb`.`Call_traffic` ;

CREATE TABLE IF NOT EXISTS `mydb`.`Call_traffic` (
  `call_ID` INT NOT NULL,
  `call_date` DATE NOT NULL,
  `call_duration` INT NOT NULL,
  `call_cost` DECIMAL(10,2) NOT NULL,
  `call_type` VARCHAR(45) NOT NULL,
  `customer_ID` INT NOT NULL,
  PRIMARY KEY (`call_ID`),
  INDEX `customer_ID_idx` (`customer_ID` ASC) VISIBLE,
  CONSTRAINT `customer_traffic_ID`
    FOREIGN KEY (`customer_ID`)
    REFERENCES `mydb`.`Customer` (`customer_ID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Customer_Plan`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `mydb`.`Customer_Plan` ;

CREATE TABLE IF NOT EXISTS `mydb`.`Customer_Plan` (
  `customer_ID` INT NOT NULL,
  `plan_ID` INT NOT NULL,
  PRIMARY KEY (`customer_ID`, `plan_ID`),
  INDEX `plan_ID_idx` (`plan_ID` ASC) VISIBLE,
  CONSTRAINT `customer_ID`
    FOREIGN KEY (`customer_ID`)
    REFERENCES `mydb`.`Customer` (`customer_ID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `plan_ID`
    FOREIGN KEY (`plan_ID`)
    REFERENCES `mydb`.`Plan` (`plan_ID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Cancellation`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `mydb`.`Cancellation` ;

CREATE TABLE IF NOT EXISTS `mydb`.`Cancellation` (
  `cancellation_ID` INT NOT NULL,
  `customer_ID` INT NOT NULL,
  `cancellation_date` DATE NOT NULL,
  `cancellation_reason` VARCHAR(500) NULL,
  PRIMARY KEY (`cancellation_ID`),
  INDEX `customer_ID_idx` (`customer_ID` ASC) VISIBLE,
  CONSTRAINT `customer_cancellationfk_ID`
    FOREIGN KEY (`customer_ID`)
    REFERENCES `mydb`.`Customer` (`customer_ID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

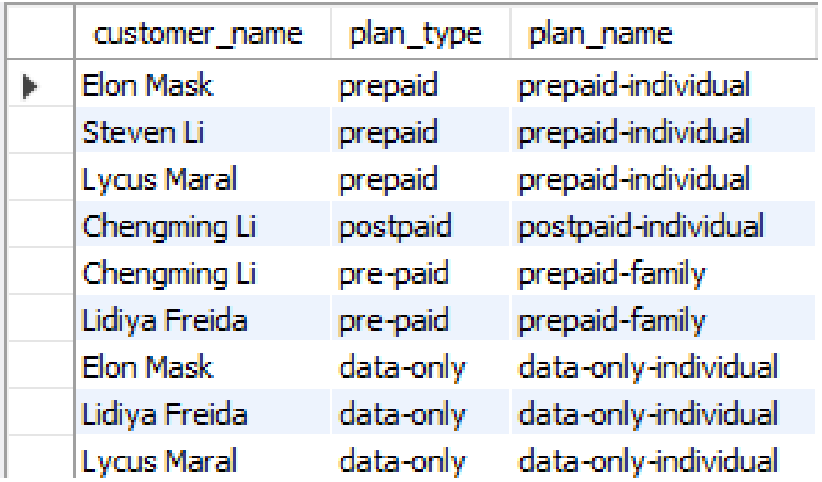
---

3. Insert Data

```
USE mydb;

-- Insert Data

INSERT INTO `customer`(`customer_ID`, `customer_name`, `customer_address`,
`customer_phone`, `customer_email`, `customer_DOB`) VALUES
('1', 'Elon Mask', '4156 Brannon Street', '(310)-720-6740', 'vibix70301@syinxun.com',
'2004-11-25') ,
('2', 'Steven Li', '2742 Glenwood Avenue', '(494) 699-7762',
'kavkaz7777@chantellegribbon.com', '2001-01-25') ,
('3', 'Chengming Li', '820 Bel Meadow Drive', '(683) 641-6204',
'jillianoreilly@naverly.com', '1999-08-23') ,
('4', 'Lidiya Freida', '3385 Michael Street', '(424) 575-4621', '986530898@dgaga.com',
'2004-01-01') ,
('5', 'Lycus Maral', '4907 Desert Broom Court', '(966) 352-2198',
'49697438@syinxun.com', '1980-07-13') ;

INSERT INTO `plan`(`plan_ID`, `plan_type`, `plan_name`, `plan_duration`, `plan_cost`,
`plan_features`) VALUES
('1', 'prepaid', 'prepaid-individual', '120', '9.99', '2G DATA and 120 mins phone call'),
('2', 'postpaid', 'postpaid-individual', '100', '9.99', '2G DATA and 100 mins phone call'),
('3', 'pre-paid', 'prepaid-family', '120', '69.99', 'unlimit DATA and 120 mins phone call for
7 people'),
('4', 'data-only', 'data-only-individual', '0', '19.99', 'unlimit DATA');

INSERT INTO `transaction`(`transaction_ID`, `customer_ID`, `plan_ID`,
`transaction_type`, `transaction_date`, `transaction_amount`) VALUES
('1', '1', '1', 'plan activation', '2021-04-26', '9.99'),
('2', '1', '1', 'plan renewal', '2021-05-26', '9.99'),
('3', '1', '4', 'plan change', '2024-04-26', '19.99'),
('4', '2', '1', 'plan activation', '2018-01-03', '9.99'),
('5', '2', '1', 'plan renewal', '2018-02-03', '9.99'),
('6', '2', '1', 'cancellation', '2018-03-03', '0.00'),
('7', '3', '2', 'plan activation', '2020-11-09', '9.99'),
('8', '3', '2', 'plan renewal', '2020-12-10', '9.99'),
('9', '3', '3', 'plan change', '2021-01-10', '69.99'),
('10', '4', '4', 'plan activation', '2020-07-01', '19.99'),
('11', '4', '4', 'plan renewal', '2020-08-10', '19.99'),
('12', '4', '3', 'plan change', '2020-09-10', '69.99'),
('13', '5', '1', 'plan activation', '2021-04-26', '9.99'),
('14', '5', '1', 'plan renewal', '2021-05-26', '9.99'),
('15', '5', '4', 'plan change', '2021-07-26', '19.99');

INSERT INTO `customer_plan`(`customer_ID`, `plan_ID`) VALUES
('1', '1'),
('1', '4'),
('2', '1'),
('3', '2'),
('3', '3'),
('4', '3'),
('4', '4'),
('5', '1'),
('5', '4');

INSERT INTO `call_traffic`(`call_ID`, `customer_ID`, `call_date`, `call_duration`,
`call_cost`,`call_type`) VALUES
('1', '1', '2015-04-26', '10', '0.00', 'local'),
('2', '1', '2018-04-26', '150', '5.00', 'local'),
('3', '1', '2023-02-26', '10', '0.00', 'national'),
('4', '2', '2015-01-06', '1', '0.00', 'local'),
('5', '2', '2018-07-26', '15', '15.00', 'international'),
('6', '2', '2020-02-13', '10', '5.00', 'national'),
('7', '3', '2015-04-09', '10', '0.00', 'local'),
('8', '3', '2018-04-26', '190', '190.00', 'international'),
('9', '3', '2019-11-16', '7', '3.50', 'national'),
('10', '4', '2019-07-26', '10', '10.00', 'international'),
('11', '4', '2019-06-28', '150', '15.00', 'local'),
('12', '4', '2020-02-26', '10', '5.00', 'national'),
('13', '5', '2021-07-26', '10', '10.00', 'international'),
('14', '5', '2021-08-08', '150', '150.00', 'international'),
('15', '5', '2021-12-16', '10', '10.00', 'international');

INSERT INTO `cancellation`(`cancellation_ID`, `customer_ID`, `cancellation_date`,
`cancellation_reason`) VALUES
('1', '2', '2018-03-03', 'choose other carrier');
```

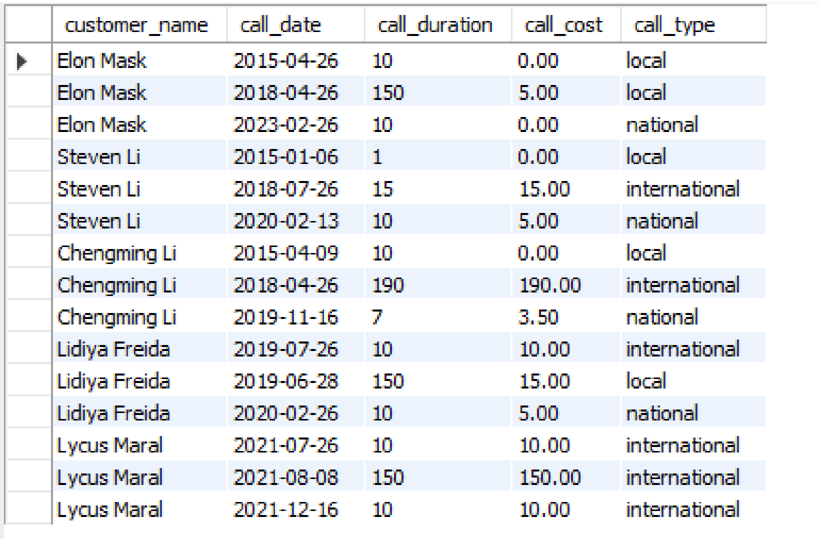
4. Query

```
-- Create a Customer Summary showing all plans for each customer, including any
cancellations

SELECT  customer_name, plan.plan_type, plan_name
FROM customer
INNER JOIN customer_plan
ON customer.customer_ID = customer_plan.customer_ID
INNER JOIN plan
ON customer_plan.plan_ID = plan.plan_ID;
```



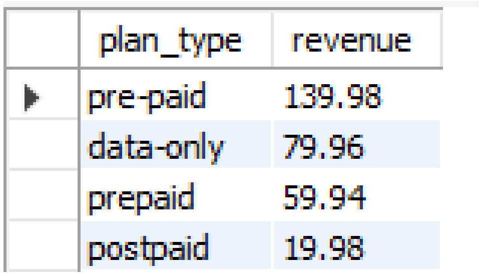
	customer_name	plan_type	plan_name
▶	Elon Mask	prepaid	prepaid-individual
	Steven Li	prepaid	prepaid-individual
	Lycus Maral	prepaid	prepaid-individual
	Chengming Li	postpaid	postpaid-individual
	Chengming Li	pre-paid	prepaid-family
	Lidiya Freida	pre-paid	prepaid-family
	Elon Mask	data-only	data-only-individual
	Lidiya Freida	data-only	data-only-individual
	Lycus Maral	data-only	data-only-individual

```
-- Create a Customer Call Log

SELECT customer_name, call_date, call_duration, call_cost, call_type
FROM customer
INNER JOIN call_traffic
ON customer.customer_ID = call_traffic.customer_ID;
```



	customer_name	call_date	call_duration	call_cost	call_type
▶	Elon Mask	2015-04-26	10	0.00	local
	Elon Mask	2018-04-26	150	5.00	local
	Elon Mask	2023-02-26	10	0.00	national
	Steven Li	2015-01-06	1	0.00	local
	Steven Li	2018-07-26	15	15.00	international
	Steven Li	2020-02-13	10	5.00	national
	Chengming Li	2015-04-09	10	0.00	local
	Chengming Li	2018-04-26	190	190.00	international
	Chengming Li	2019-11-16	7	3.50	national
	Lidiya Freida	2019-07-26	10	10.00	international
	Lidiya Freida	2019-06-28	150	15.00	local
	Lidiya Freida	2020-02-26	10	5.00	national
	Lycus Maral	2021-07-26	10	10.00	international
	Lycus Maral	2021-08-08	150	150.00	international
	Lycus Maral	2021-12-16	10	10.00	international

```
-- Create a Plan Revenue report showing revenue for each plan

SELECT plan_type, SUM(transaction.transaction_amount) AS revenue
FROM plan
INNER JOIN transaction
ON plan.plan_ID = transaction.plan_ID
GROUP BY plan_type
ORDER BY SUM(transaction.transaction_amount)
DESC;
```



	plan_type	revenue
▶	pre-paid	139.98
	data-only	79.96
	prepaid	59.94
	postpaid	19.98