## DBMS Assignment 4

**MIS – 642303019**

**Statement: Create the schema and constraints on the relations**.

**Modify the trains schema which we saw earlier to create constraints to check the following:**

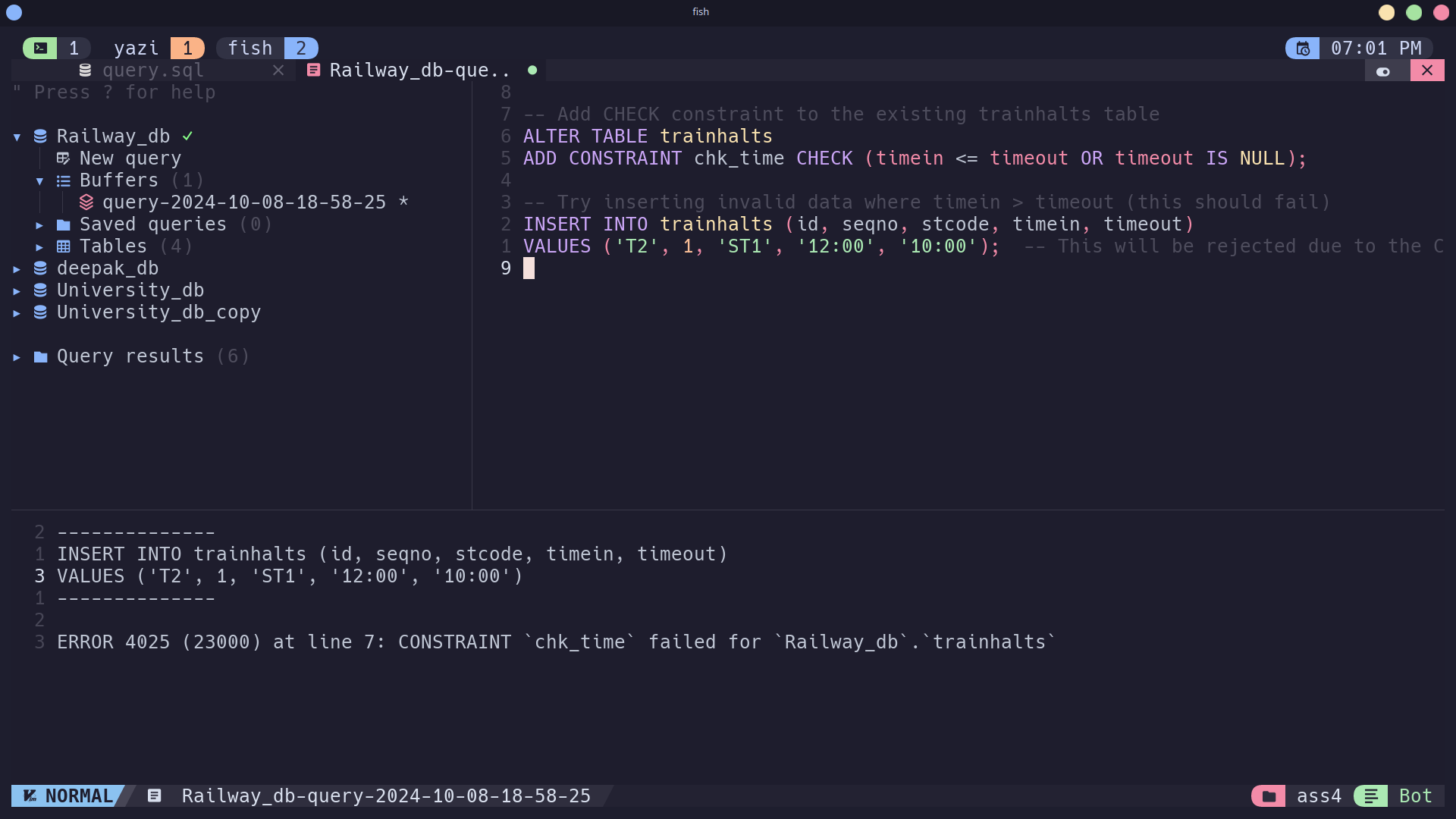
* **The value of timein is always less than or equal to timeout**

ALTER TABLE trainhalts

ADD CONSTRAINT chk\_time CHECK (timein <= timeout OR timeout IS NULL);

INSERT INTO trainhalts (id, seqno, stcode, timein, timeout)

VALUES ('T2', 1, 'ST1', '12:00', '10:00');



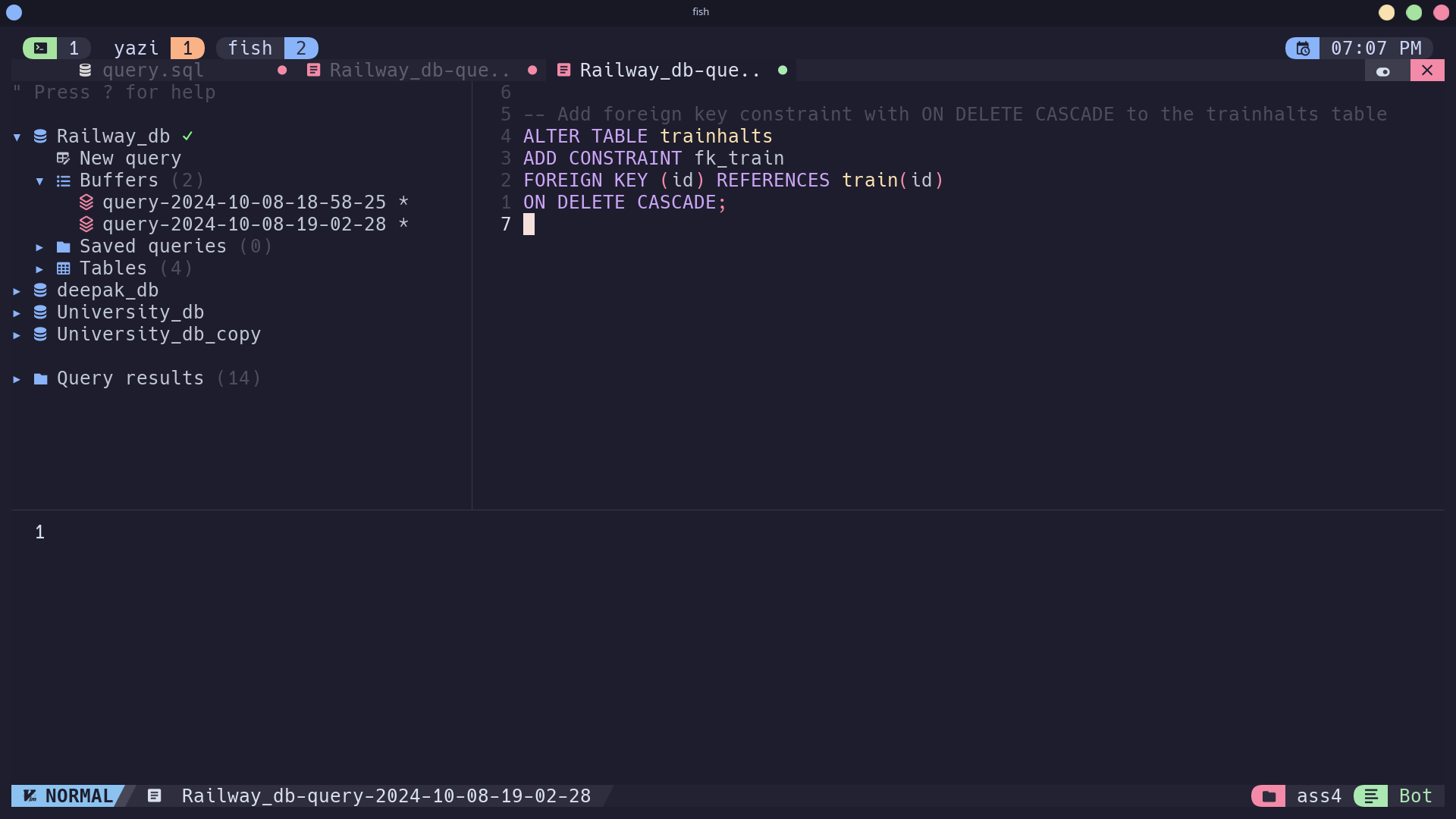
* **When a train is removed from service, all its halts should be deleted**.

ALTER TABLE trainhalts

ADD CONSTRAINT fk\_train

FOREIGN KEY (id) REFERENCES train(id)

ON DELETE CASCADE;



* **Insert inconsistent data and verify the constraints**.

-- Insert valid train and halts data

INSERT INTO train (id, name) VALUES ('T2', 'Superfast Express');

-- Add halts for the train

INSERT INTO trainhalts (id, seqno, stcode, timein, timeout)

VALUES ('T2', 0, 'ST1', NULL, '09:00'),

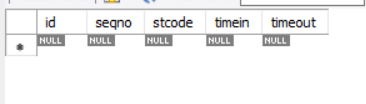
('T2', 1, 'ST2', '10:00', '10:30');

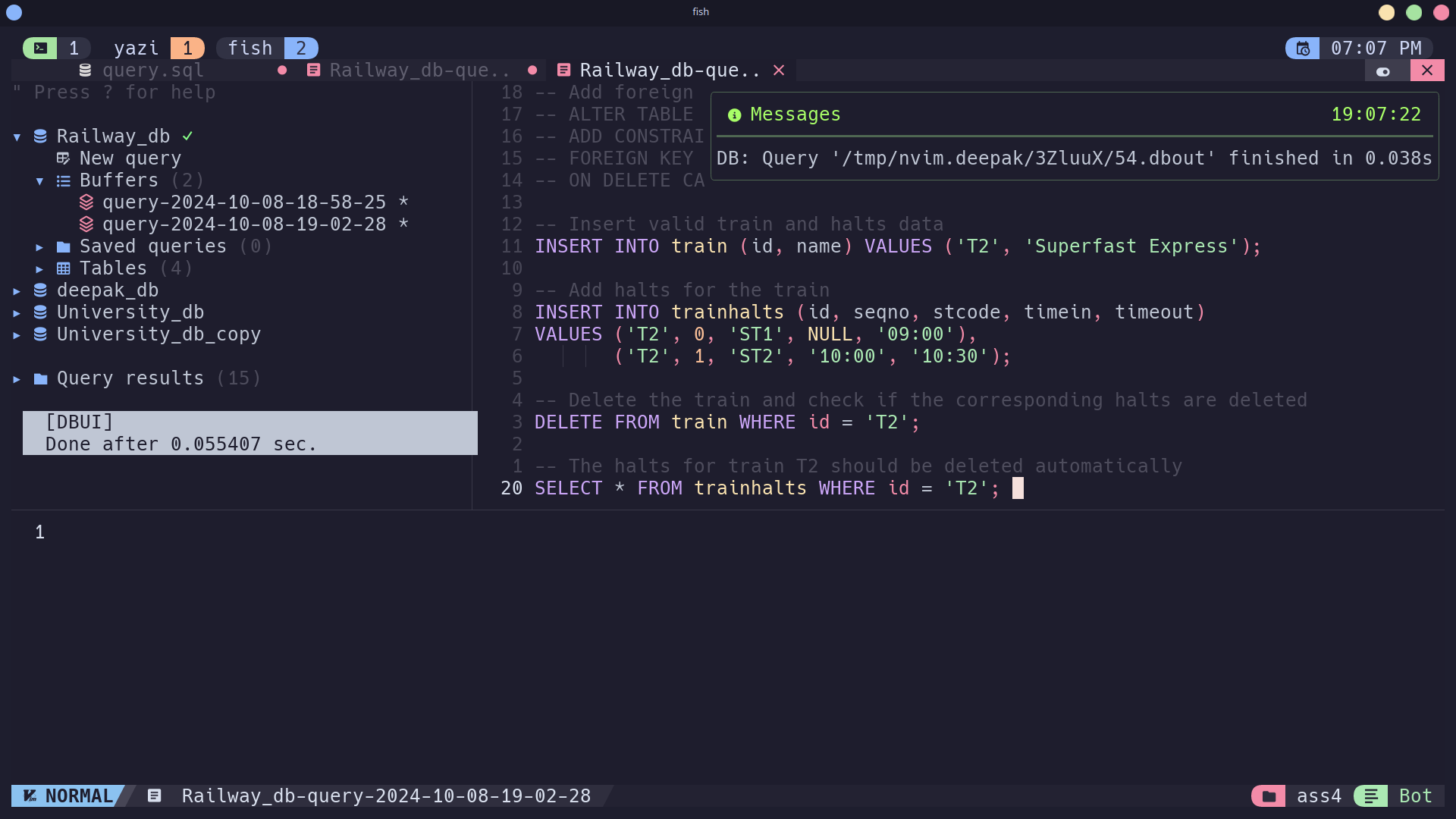
-- Delete the train and check if the corresponding halts are deleted

DELETE FROM train WHERE id = 'T2';

-- The halts for train T2 should be deleted automatically

SELECT \* FROM trainhalts WHERE id = 'T2';





Write SQL Create table statements to create the following schema. Include all appropriate primary and foreign key declarations. Choose appropriate types for each attribute.

* + remotecentre(centreId, college, town, state)
  + person(ID, name, email)
  + programme(progId, title, fromdate, todate)
  + coordinator(ID, progId, centreId)
  + participant(ID, progId, centreId)

CREATE TABLE remotecentre (

centreId INT PRIMARY KEY, -- Unique ID for the remote centre

college VARCHAR(100), -- Name of the college

town VARCHAR(50), -- Name of the town

state VARCHAR(50) -- Name of the state

);

CREATE TABLE person (

ID INT PRIMARY KEY, -- Unique person ID

name VARCHAR(100), -- Person's name

email VARCHAR(100) -- Person's email

);

CREATE TABLE programme (

progId INT PRIMARY KEY, -- Unique programme ID

title VARCHAR(100), -- Title of the programme

fromdate DATE, -- Start date of the programme

todate DATE -- End date of the programme

);

CREATE TABLE coordinator (

ID INT, -- ID of the coordinator (Person ID)

progId INT, -- ID of the programme

centreId INT, -- ID of the remote centre

PRIMARY KEY (ID, progId, centreId), -- Composite primary key

FOREIGN KEY (ID) REFERENCES person(ID), -- Foreign key to the person table

FOREIGN KEY (progId) REFERENCES programme(progId), -- Foreign key to the programme table

FOREIGN KEY (centreId) REFERENCES remotecentre(centreId) -- Foreign key to the remotecentre table

);

CREATE TABLE participant (

ID INT,

progId INT,

centreId INT,

PRIMARY KEY (ID, progId, centreId)

FOREIGN KEY (ID) REFERENCES person(ID),

FOREIGN KEY (progId) REFERENCES programme(progId),

FOREIGN KEY (centreId) REFERENCES remotecentre(centreId)

);







