CS2102

Tutorial 06

Question 2

- Make a trigger on customers table to check that customers cannot be workers (or symmetrically, make a trigger on workers table to check that workers cannot be customers).
- Check query?
  -
- Code structure?
  -

Question 2

- Make a trigger on customers table to check that customers cannot be workers (or symmetrically, make a trigger on workers table to check that workers cannot be customers).
- Check query?

  -
    ```
    SELECT COUNT(*) > 0 FROM Workers W
    WHERE  NEW.uname = W.uname;
    ```
- Code structure?
  - `IF check THEN RETURN NULL; ELSE RETURN NEW; END IF`
    - Prevent or allow insertion based on check

Question 3

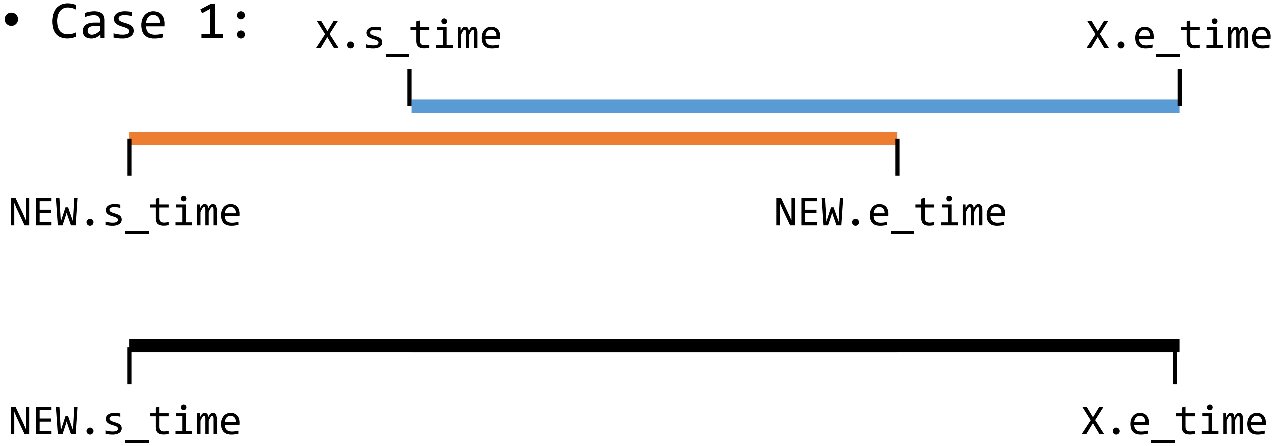
- Create a stored procedure add_customers that takes in uname and pass and add the customer in a way that satisfy covering constraint on the ISA (or alternatively, create a stored procedure add_workers that takes in uname, pass, and expr and add the customer in a way that satisfy covering constraint on the ISA).
- What needs to be modified?
  - Customers table
  - Users table

Question 3

- Create a stored procedure add_customers that takes in uname and pass and add the customer in a way that satisfy covering constraint on the ISA (or alternatively, create a stored procedure add_workers that takes in uname, pass, and expr and add the customer in a way that satisfy covering constraint on the ISA).
- What needs to be modified?
    - Customers table
        - `INSERT INTO Users VALUES (uname, pass);`
    - Users table
        - `INSERT INTO Customers VALUES (uname);`

Question 4

- Create a trigger on any update and insertion on availability to check for overlap.
- How to check if overlap?
  - Case analysis
    - Case 1:



X.s_time

X.e_time

NEW.s_time

NEW.e_time

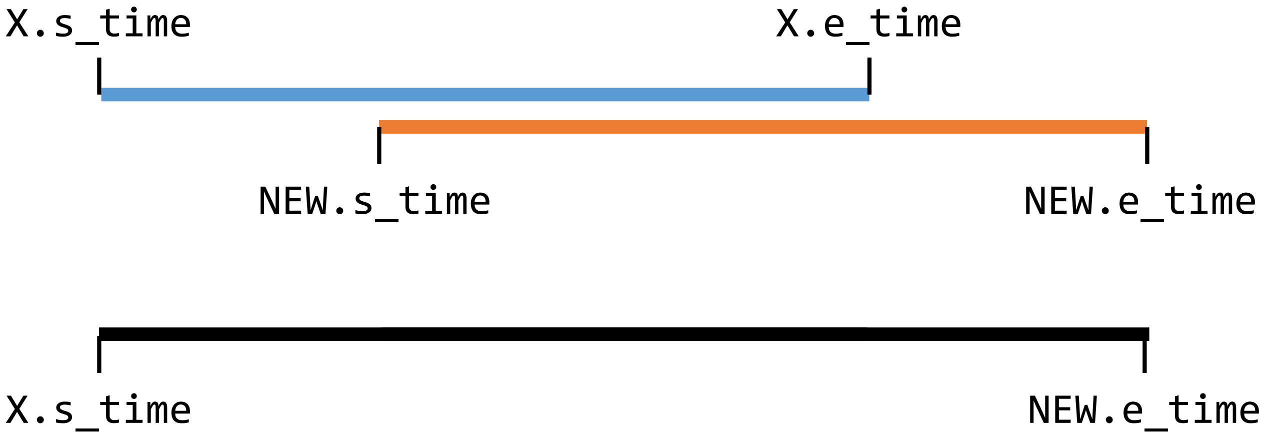
NEW.s_time

X.e_time

NEW.s_time < X.s_time < NEW.e_time < X.e_time ==> (NEW.s_time, X.e_time)

Question 4

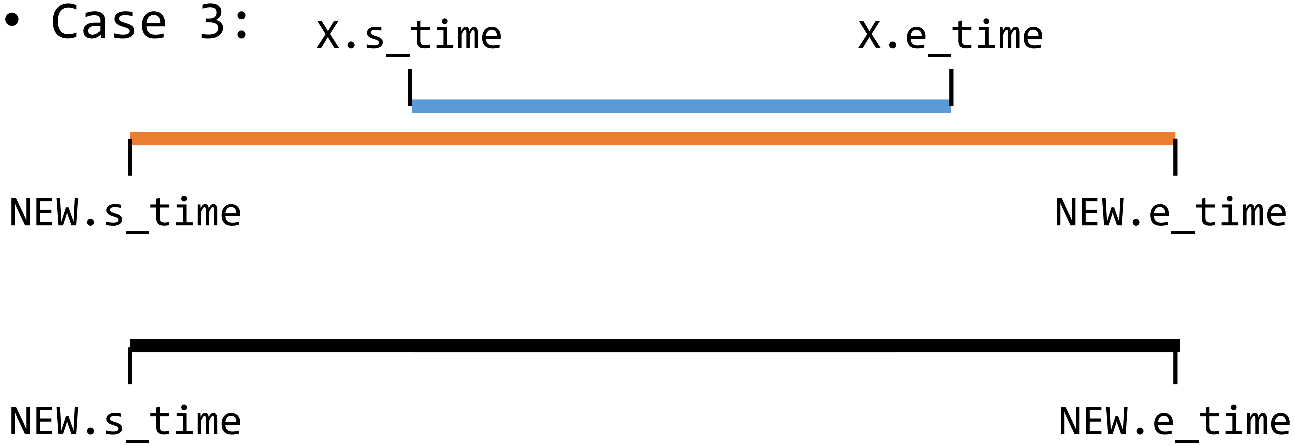
- Create a trigger on any update and insertion on availability to check for overlap.
- How to check if overlap?
  - Case analysis
    - Case 2:



```
X.s_time < NEW.s_time < X.e_time < NEW.e_time  ==>  (X.s_time, NEW.e_time)
```

Question 4

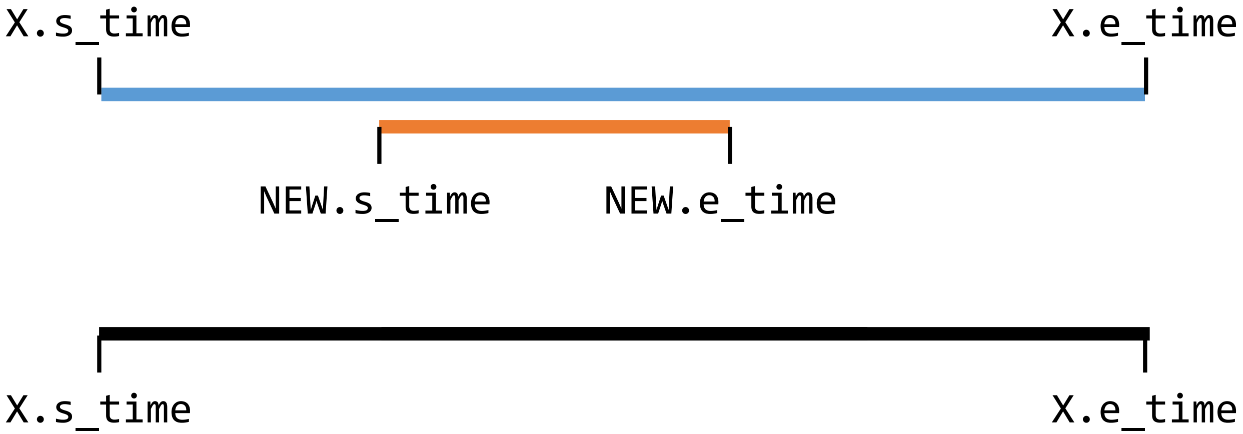
- Create a trigger on any update and insertion on availability to check for overlap.
- How to check if overlap?
  - Case analysis
    - Case 3:



NEW.s_time < X.s_time < X.e_time < NEW.e_time ==> (NEW.s_time, NEW.e_time)

Question 4

- Create a trigger on any update and insertion on availability to check for overlap.
- How to check if overlap?
  - Case analysis
    - Case 4:



```
X.s_time < NEW.s_time < NEW.e_time < X.e_time  ==>  (X.s_time, S.e_time)
```

Question 5

- We want to choose the manager with the highest past work experience that has not become a manager.
- Finding the manager?

  -
    ```
    SELECT   uname FROM Workers W
    WHERE    uname NOT IN (SELECT uname FROM Manages)
    ORDER BY expr DESC, uname ASC
    LIMIT    1
    ```

Question 6

- We wish to prevent bidding on any availability that already has a winner. This can be enforced via trigger on any insertion or update to bid. Create this trigger.
    - Checking the condition?
        -
        -
    - Code structure
        -

Question 6

- We wish to prevent bidding on any availability that already has a winner. This can be enforced via trigger on any insertion or update to bid. Create this trigger.
  - Checking the condition?

    -
      ```
      SELECT 1 FROM Bid B
      WHERE  NEW.ctuname = B.ctuname AND NEW.s_date = B.s_date
        AND  NEW.s_time  = B.s_time  AND NEW.e_time = B.e_time
        AND  B.is_win;
      ```
    - 1 if exists, null if not exists
  - Code structure
    - `IF check THEN RETURN NULL; ELSE RETURN NEW; END IF`

Question 7

- We only allow worker to train care takers if they have a combined total past work experience and current working experience at PetER of 5 years.
- Checking the condition?

  -
    ```
    SELECT DISTINCT uname FROM Workers W
    WHERE  (SELECT EXTRACT(AGE(NEW.s_date, W.s_date)))
           >
           5-W.expr
    ```