**Trigger 1**

/\* 1.

Stops a waitstaff employee from being assigned to more than 1 section\*/

DROP TRIGGER IF EXISTS `323termproject`.`wait\_staff\_BEFORE\_INSERT`;

DELIMITER $$

USE `323termproject`$$

CREATE DEFINER=`root`@`localhost` TRIGGER `wait\_staff\_BEFORE\_INSERT` BEFORE INSERT ON `wait\_staff` FOR EACH ROW

BEGIN

if exists( select empID from wait\_staff where empID = new.empID) THEN

signal sqlstate '45000' set message\_text = 'Wait Staff can only be assigned to one section';

end if;

END$$

DELIMITER ;

**Business Rule 1 failure**

13:03:28 insert into wait\_staff(sectionID, empID) values (1, 3)

Error Code: 1644. Wait Staff can only be assigned to one section 0.000 sec

**Business Rule 1 success**

13:04:28 insert into wait\_staff(sectionID, empID) values (2, 3), (1, 11)

2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0 0.016 sec

**Trigger 2**

/\* 2.

Stops the added section from having more than 5 tables\*/

DROP TRIGGER IF EXISTS `323termproject`.`section\_BEFORE\_INSERT`;

DELIMITER $$

USE `323termproject`$$

CREATE DEFINER=`root`@`localhost` TRIGGER `section\_BEFORE\_INSERT` BEFORE INSERT ON `section` FOR EACH ROW

BEGIN

if new.numTables > 5 THEN

signal sqlstate '45000' set message\_text = 'Each section can have at most 5 tables';

end if;

END$$

DELIMITER ;

**Business Rule 2 failure**

13:14:58 insert into section(sectionID,numTables, maxPpl) values (3, 6, 4)

Error Code: 1644. Each section can have at most 5 tables 0.000 sec

**Business Rule 2 success**

13:14:50 insert into `Table` (tableID, sectionID ,capacity) values (11, 1, 2), (12, 1, 2), (21, 2, 4), (22, 2, 4)

4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0 0.000 sec

**Trigger 3**

/\* 3.

Stops a station from having more than 2 line cooks when a new cook is added\*/

DROP TRIGGER IF EXISTS `323termproject`.`station\_assignment\_BEFORE\_INSERT`;

DELIMITER $$

USE `323termproject`$$

CREATE DEFINER=`root`@`localhost` TRIGGER `station\_assignment\_BEFORE\_INSERT` BEFORE INSERT ON `station\_assignment` FOR EACH ROW

BEGIN

if (select count(station\_assignment.empID) from station\_assignment where new.station\_name = station\_name) > 2 THEN

signal sqlstate '45000' set message\_text = 'Each station can have at most 2 cooks';

end if;

END$$

DELIMITER ;

**Business Rule 3 failure**

13:44:51 insert into station\_assignment(station\_name, empID) values ('Salad Station', 21)

Error Code: 1644. Each station can have at most 2 cooks 0.000 sec

**Business Rule 3 success**

13:14:50 insert into station\_assignment(station\_name, empID) values ('Salad Station', 6), ('Soup Station', 6), ('Entree Station', 6)

3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 0.016 sec

**Trigger 4**

/\* 4.

Stops the insertion of a new station assignment for a cook that already has the maximum of 3 stations

assigned to them\*/

DROP TRIGGER IF EXISTS `323termproject`.`station\_assignment\_BEFORE\_INSERT`;

DELIMITER $$

USE `323termproject`$$

CREATE DEFINER=`root`@`localhost` TRIGGER `station\_assignment\_BEFORE\_INSERT` BEFORE INSERT ON `station\_assignment` FOR EACH ROW BEGIN

if (select count(empID) from station\_assignment where empID = new.empID) > 3 THEN

signal sqlstate '45000' set message\_text = 'Each line cook can have at most 3 stations assigned to them';

end if;

END$$

DELIMITER ;

**Business Rule 4 failure**

13:44:51 insert into station\_assignment(station\_name, empID) values ('Salad Station', 21)

Error Code: 1644. Each line cook can have at most 3 stations assigned to them 0.000 sec

**Business Rule 4 success**

13:14:50 insert into station\_assignment(station\_name, empID) values ('Salad Station', 6), ('Soup Station', 6), ('Entree Station', 6)

3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 0.016 sec

**Trigger 5**

/\* 5.

Stops the insertion of a scheduled shift without having at least 1 maitre d\*/

DROP TRIGGER IF EXISTS `323termproject`.`scheduled\_shift\_BEFORE\_INSERT`;

DELIMITER $$

USE `323termproject`$$

CREATE DEFINER = CURRENT\_USER TRIGGER `323termproject`.`scheduled\_shift\_BEFORE\_INSERT` BEFORE INSERT ON `scheduled\_shift` FOR EACH ROW

BEGIN

if exists(

select count(empID)

from scheduled\_shift natural join employee natural join part\_time natural join maitre\_d

where scheduled\_shift.empID = maitre\_d.empID

group by empID

having count(empID) < 1)

THEN

signal sqlstate '45000' set message\_text = 'There must be at least 1 maitre d per shift';

end if;

END$$

DELIMITER ;

/\*Stops the deletion of a scheduled shift that would lead to there not being at least 1 maitre d per shift\*/

DROP TRIGGER IF EXISTS `323termproject`.`scheduled\_shift\_BEFORE\_DELETE`;

DELIMITER $$

USE `323termproject`$$

CREATE DEFINER = CURRENT\_USER TRIGGER `323termproject`.`scheduled\_shift\_BEFORE\_DELETE` BEFORE DELETE ON `scheduled\_shift` FOR EACH ROW

BEGIN

if exists(

select count(empID)

from scheduled\_shift natural join employee natural join part\_time natural join maitre\_d

where scheduled\_shift.empID = maitre\_d.empID and OLD.empID = maitre\_d.empID and old.empID = scheduled\_shift.empID

group by empID

having count(empID) = 1)

THEN

signal sqlstate '45000' set message\_text = 'There must be at least 1 maitre d per shift';

end if;

END$$

DELIMITER ;

**Business Rule 5 failure**

13:44:52 insert into `scheduled\_shift`(`empID`, `shift\_date`, `start`, `end`) values (12, '2020-03-04', '8:00:00', '12:00:00')

Error Code: 1644. There must be at least 1 maitre d per shift 0.000 sec

**Business Rule 5 success**

13:44:47 insert into `scheduled\_shift`(`empID`, `shift\_date`, `start`, `end`) values (12, '2020-03-04', '8:00:00', '12:00:00'), (12, '2020-03-04', '12:00:00', '17:00:00'), (12, '2020-03-04', '17:00:00', '21:00:00'), (1, '2020-03-04', '8:00:00', '12:00:00'), (3, '2020-03-04', '12:00:00', '17:00:00'), (2, '2020-03-04', '17:00:00', '21:00:00'), (7, '2020-03-04', '17:00:00', '21:00:00')

7 row(s) affected Records: 7 Duplicates: 0 Warnings: 0 0.000 sec