1. Створити таблицю-журнал тригер, на видалення даних із таблиці foods (реєструвати назву блюда, що видаляється). Додати та потім видалити рядок із таблиці foods для пеевірки роботи тригера.

CREATE TRIGGER deleted\_food\_report

AFTER DELETE

ON foods

FOR EACH ROW

BEGIN

INSERT INTO deleted\_foods (

food\_name

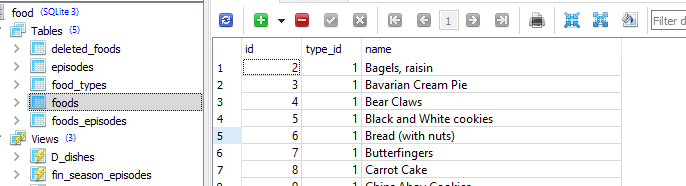
)

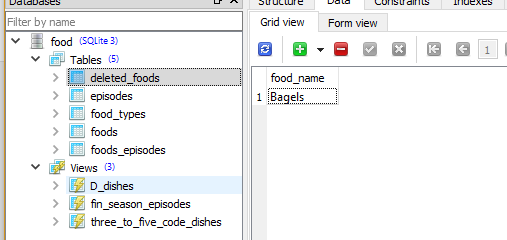
VALUES (

OLD.name

);

END;





1. Створити тригер, що при видаленні серії перевіряє, щоб це була не остання серія у сезоні та якщо так, відміняє операцію.

CREATE TRIGGER check\_last\_episode\_delete

BEFORE DELETE

ON episodes

FOR EACH ROW

BEGIN

SELECT CASE WHEN (

SELECT t1.season

FROM episodes AS t1,

episodes AS t2

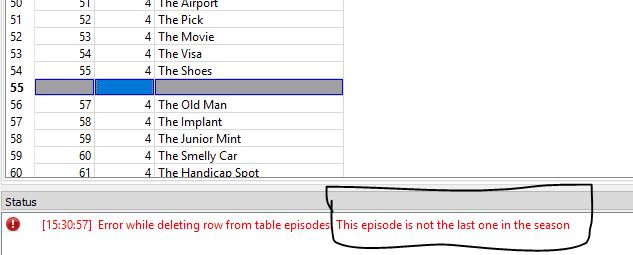
WHERE (t1.season) = (t2.season)

)

THEN RAISE(ABORT, "This episode is not the last one in the season")

END;

END;



1. Створити тригер, що забороняє зміну поля сезон таблички епізодів.

CREATE TRIGGER forbid\_update\_season\_field

BEFORE UPDATE

ON episodes

BEGIN

SELECT RAISE(ABORT, "YOU ARE NOT ALLOWED TO MODIFY THIS COLUMN") END;

END;

Має працювати скрипт нижче, але його не сприймає консоль студії

CREATE TRIGGER forbid\_update\_season\_field

BEFORE UPDATE OF season

ON episodes

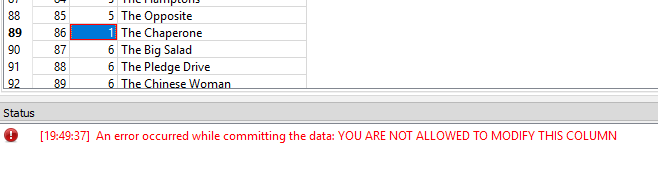
FOR EACH ROW

BEGIN

SELECT RAISE(ABORT, "YOU ARE NOT ALLOWED TO MODIFY THIS COLUMN")

END;

END;



1. Створити тригер, що забороняє додавати більше 12 різних сезонів (при чому номер сезону може біти >12, наприклад 1, 2 та 20 сезони – всього їх 3, отже вимога не порушується)

CREATE TRIGGER seasons\_limit

BEFORE INSERT

ON episodes

FOR EACH ROW

BEGIN

SELECT CASE WHEN (

SELECT COUNT(DISTINCT season)

FROM episodes

)

> 12 THEN RAISE(ABORT, "Cannot add more than 12 seasons")

END;

END;



1. Створити 3 тригера (по одному додавання, зміна та видалення) до одної з табличок із свого варіанта індивідуального завдання. Зміст тригера задати за власним бажанням.

Видалення

DROP TRIGGER random\_delete;

CREATE TRIGGER random\_delete

BEFORE DELETE

ON foods

BEGIN

SELECT CASE WHEN (

SELECT t1.type\_id

FROM foods AS t1,

foods AS t2

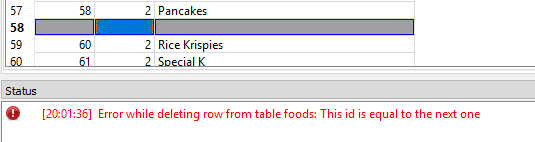
WHERE (t1.type\_id) = (t2.type\_id)

)

THEN RAISE(ABORT, "This id is equal to the next one")

END;

END;



Зміна

CREATE TRIGGER random\_change

BEFORE UPDATE

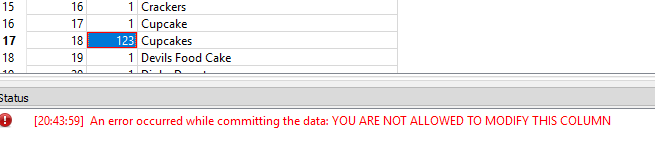
ON foods

BEGIN

SELECT RAISE(ABORT, "YOU ARE NOT ALLOWED TO MODIFY THIS COLUMN")

END;

END;



Додавання

CREATE TRIGGER random\_add

BEFORE INSERT

ON foods

FOR EACH ROW

BEGIN

SELECT CASE WHEN (

SELECT COUNT(DISTINCT type\_id)

FROM foods

)

> 15 THEN RAISE(ABORT, "Cannot add more than 15 types")

END;

END;

