**One Less Crochet**

**Company Overview:**

At One Less Crochet, we believe in the power of creativity and craftsmanship, and we're on a mission to make the world a better place, one stitch at a time. Our carefully curated selection of crochet products showcases the finest and most eco-friendly materials available. From organic cotton and bamboo yarns to recycled fibers, we are committed to reducing our carbon footprint and promoting eco-conscious choices in the crafting community.

* **Craft ID:** A unique identifier assigned to every "Craft" product for inventory and tracking purposes.
* **Craft Name:** The name or title given to the specific "Craft" product, helping customers easily identify and refer to it.
* **Selling Price:** The retail price at which the "Craft" product is offered to customers.
* **Manufacturing Cost:** The total cost incurred during the production of the "Craft" item, including materials, labor, and other production expenses.
* **Size:** The dimensions or measurements of the "Craft" product, providing customers with information about its physical size.
* **Color:** The color or color variants available for the "Craft" product, allowing customers to select their preferred option.
* **Type:** The size variations available for the "Craft" product. The values can be XS, S, M, L, and XL, indicating different size options suitable for various crafting projects.
* **Fit:** If applicable, the "Fit" attribute indicates the compatibility of the "Craft" product with specific crafting materials or other accessories.
* **Description:** A detailed and engaging description of the "Craft" product, highlighting its features, benefits, and eco-friendly nature. This description would entice customers to choose "Craft" and provide essential information about its use and purpose in crafting projects.

By carefully defining these attributes for the "Craft" product, One Less Craft can efficiently manage its inventory, assist customers in making informed decisions, and showcase the product's unique selling points, ultimately enhancing the overall customer experience.

Top of Form

**Product: Crafts**

**Product Attributes:**

|  |  |
| --- | --- |
| **Attributes** | **Sample Data** |
| Craft ID | 100001 |
| Craft Name | Cushion Macrame |
| Selling Price | 43.97 |
| Manufacturing Cost | 15.64 |
| Size | M |
| Color | Beige |
| Type | Macrame |
| Fit | Null |
| Description | This beautiful woven cushion is ideal to add some boho chic to your home. Tied using our most sustainable, recycled cotton yarn Nova Vita 4 in a calming neutral cream shade. |

**One Less Craft**

347 Kings Street

Toronto, ON M1C 2J8

647-876-3589

**Customer Sales Invoice**

**Customer ID**: 000123 **Order ID**: 09101

**Customer Name**: Stephen H. **Order Date**: 02/10/2019

**Customer Address**: 1070 Sheppard Ave **Associate ID**: 501

Toronto, ON M4K 5H8 **Associate Name**: John Doe

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| **Product ID** | **Name** | **Size** | **Fit** | **Color** | **Description** | **Qty** | **Unit Price** | **Extended Price** |
| LB171 | 62-12345 | M |  |  | Cordless Electric Lawn Mower | 2 | $595.00 | $1190.00 |
| BD518 | 59-98765 | S |  |  | Riding Lawn Mower | 1 | $1449.00 | $1449.00 |
| CM713 | 36-59278 | XL |  |  | Self-Propelled Gas Push Mower | 2 | $625.00 | $1250.00 |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  | **Sub-total** | $3889.00 |
|  |  |  |  |  |  |  | **Tax** | $505.57 |
|  |  |  |  |  |  | **Invoice Total** | | $4,394.57 |

**Entity Relationship Diagram**

**Many To Many Relationships**

A black screen with white text

Description automatically generated

**Entity Relationship Diagram (Many to Many relationships resolved)**

A diagram of a computer system

Description automatically generated with medium confidence

**ERD Language**

Each EMPLOYEE *must* work for zero or one DEPARTMENT.

Each DEPARTMENT *may* employee zero, one or more EMPLOYEEs.

Each EMPLOYEE *may* supervise zero or one DEPARTMENT.

Each DEPARTMENT *may* be supervised by zero or one EMPLOYEE.

Each EMPLOYEE (Mgr. role) may supervise zero, one or more EMPLOYEEs (Emp role).

Each EMPLOYEE (Emp role) may be supervised by zero or one EMPLOYEE (Mgr. role).

Each EMPLOYEE *must* work for one and only one STORE.

Each STORE *must* employee one or more EMPLOYEEs.

Each EMPLOYEE *may* supervise one and only one STORE.

Each STORE *may* be supervised by one and only one EMPLOYEE.

Each EMPLOYEE *may* be included in zero, one or more ORDERs.

Each ORDER *must* have one and only one EMPLOYEE.

Each EMPLOYEE *may* hold one and only one JOB.

Each JOB *may* be held by zero, one or more EMPLOYEEs.

Each EMPLOYEE *may* serve zero, one or more CUSTOMERs.

Each CUSTOMER *may* be served by zero or one EMPLOYEE.

Each CUSTOMER *must* have one and only one ZIP.

Each CUSTOMER *may* prefer zero or one CRAFT.

Each CRAFT *may* be preferred by zero, one or more CUSTOMERs.

Each ZIP *may* belong to zero, one or more CUSTOMERs.

Each ZIP *may* belong to zero, one or more WAREHOUSEs.

Each WAREHOUSE *must* have one and only one ZIP.

Each ZIP *may* belong to zero, one or more STOREs.

Each STORE *must* have one and only one ZIP.

Each CUSTOMER *may* be included in zero, one or more ORDERs.

Each ORDER *must* have one and only one CUSTOMER.

Each ORDER *may* have zero, one or more ORDER\_CRAFTs.

Each ORDER\_CRAFT *must* be for one and only one ORDER.

Each CRAFT *may* have zero, one or more ORDER\_CRAFTs.

Each ORDER\_CRAFT *must* be for one and only one CRAFT.

Each CRAFT *may* have zero, one or more CRAFT\_YARNs.

Each CRAFT\_YARN *must* be for one and only one CRAFT.

Each YARN *may* have zero, one or more CRAFT\_YARNs.

Each CRAFT\_YARN *must* be for one and only one YARN.

Each CRAFT *may* have zero, one or more INVENTORYs.

Each INVENTORY *must* be for one and only one CRAFT.

Each WAREHOUSE *may* have zero, one or more INVENTORYs.

Each INVENTORY *must* be for one and only one WAREHOUSE.

Each WAREHOUSE *may* have zero one or more WAREHOUSE\_STOREs.

Each WAREHOUSE\_STORE *must* be for one and only one WAREHOUSE.

Each STORE *may* have zero one or more WAREHOUSE\_STOREs.

Each WAREHOUSE\_STORE *must* be for one and only one STORE.

Each YARN *may* be supplied by zero, one or more YARN\_SUPPLIERs.

Each YARN\_SUPPLIER *must* be for one and only one YARN.

Each SUPPLIER *may* be supply zero, one or more YARN\_SUPPLIER.

Each YARN\_SUPPLIER *must* be for one and only one SUPPLIER.

**Relational Data Model (Physical DM)**









**DDL Statements**

CREATE SCHEMA one\_less\_craft;

-- -------------------------------------------------------------------------

TABLES

-- -------------------------------------------------------------------------

CREATE TABLE crafts (

craft\_id NUMERIC(6,0) NOT NULL,

craft\_name VARCHAR(20) NOT NULL,

selling\_price NUMERIC(9,2) NOT NULL,

manufacturing\_cost NUMERIC(9,2) NOT NULL,

size VARCHAR(4) NOT NULL,

color VARCHAR(20),

craft\_type VARCHAR(20) NOT NULL,

fit VARCHAR(20) NOT NULL,

craft\_description VARCHAR(100));

-- Primary Key

ALTER TABLE crafts

ADD CONSTRAINT crafts\_craft\_id\_pk

PRIMARY KEY (craft\_id);

-- Constraints

ALTER TABLE crafts

ADD CONSTRAINT crafts\_selling\_price\_gt\_manufacturing\_cost

CHECK (selling\_price > manufacturing\_cost);

ALTER TABLE crafts

ADD CONSTRAINT crafts\_size

CHECK (size IN ('xs', 's', 'm', 'l', 'xl'));

-- -------------------------------------------------------------------------

CREATE TABLE customers (

customer\_id NUMERIC(6,0) NOT NULL,

first\_name VARCHAR(20) NOT NULL,

last\_name VARCHAR(20) NOT NULL,

date\_of\_birth DATE NOT NULL,

email VARCHAR(50) NOT NULL,

gender CHAR(1),

contact\_number VARCHAR(10) NOT NULL,

address VARCHAR(50) NOT NULL,

shipping\_address VARCHAR(50) NOT NULL,

billing\_address VARCHAR(50),

favorite\_craft NUMERIC(6,0),

zip\_code VARCHAR(10) NOT NULL,

preferred\_sales\_rep NUMERIC(6,0));

-- Primary Key

ALTER TABLE customers

ADD CONSTRAINT customers\_customer\_id\_pk

PRIMARY KEY (customer\_id);

-- Unique Keys

ALTER TABLE customers

ADD CONSTRAINT customers\_email\_uk

UNIQUE KEY (email);

ALTER TABLE customers

ADD CONSTRAINT customers\_contact\_number\_uk

UNIQUE KEY (contact\_number);

-- Constraints

ALTER TABLE customers

ADD CONSTRAINT customers\_email

CHECK (email LIKE '%@%.%');

ALTER TABLE customers

ADD CONSTRAINT customer\_gender

CHECK (gender IN ('M', 'F'));

-- -------------------------------------------------------------------------

CREATE TABLE departments (

department\_id NUMERIC(6,0) NOT NULL,

department\_name VARCHAR(30) NOT NULL,

department\_phone VARCHAR(10) NOT NULL,

department\_email VARCHAR(50) NOT NULL,

department\_budget NUMERIC(9,2) NOT NULL,

department\_description VARCHAR(100),

manager\_id NUMERIC(6,0));

-- Primary Key

ALTER TABLE departments

ADD CONSTRAINT departments\_department\_id\_pk

PRIMARY KEY (department\_id);

-- Unique Keys

ALTER TABLE departments

ADD CONSTRAINT departments\_department\_name\_uk

UNIQUE (department\_name);

ALTER TABLE departments

ADD CONSTRAINT departments\_department\_phone\_uk

UNIQUE (department\_phone);

ALTER TABLE departments

ADD CONSTRAINT departments\_department\_email\_uk

UNIQUE (department\_email);

-- Constraints

ALTER TABLE departments

ADD CONSTRAINT departments\_department\_email

CHECK (department\_email LIKE '%@%.%');

ALTER TABLE departments

ADD CONSTRAINT departments\_department\_budget

CHECK (department\_budget >= 100000 AND department\_budget <= 1000000);

-- -------------------------------------------------------------------------

CREATE TABLE employees (

employee\_id NUMERIC(6,0) NOT NULL,

first\_name VARCHAR(20) NOT NULL,

last\_name VARCHAR(20) NOT NULL,

date\_of\_birth DATE NOT NULL,

address VARCHAR(50),

contact\_number NUMERIC(10,0) NOT NULL,

email VARCHAR(50) NOT NULL,

hire\_date DATE DEFAULT (CURRENT\_DATE),

salary NUMERIC(9,2) NOT NULL,

emp\_status VARCHAR(15) NOT NULL,

manager\_id NUMERIC(6,0),

store\_id NUMERIC(6,0),

department\_id NUMERIC(6,0),

job\_id NUMERIC(6,0));

-- Primary Key

ALTER TABLE employees

ADD CONSTRAINT employees\_employee\_id\_pk

PRIMARY KEY (employee\_id);

-- Unique Key

ALTER TABLE employees

ADD CONSTRAINT employees\_contact\_number\_uk

UNIQUE KEY (contact\_number);

ALTER TABLE employees

ADD CONSTRAINT employees\_email\_uk

UNIQUE KEY (email);

-- Constraints

ALTER TABLE employees

ADD CONSTRAINT employees\_salary\_range

CHECK (salary >= 20000 AND salary <= 200000);

ALTER TABLE employees

ADD CONSTRAINT employees\_emp\_status

CHECK (emp\_status IN ('Active', 'Inactive', 'Retired', 'On Vacation'));

ALTER TABLE employees

ADD CONSTRAINT employees\_hire\_date\_gt\_date\_of\_birth

CHECK (hire\_date > date\_of\_birth);

ALTER TABLE employees

ADD CONSTRAINT employees\_email

CHECK (email LIKE '%@%.%');

-- -------------------------------------------------------------------------

CREATE TABLE jobs (

job\_id NUMERIC(6,0) NOT NULL,

job\_title VARCHAR(30) NOT NULL);

-- Primary Key

ALTER TABLE jobs

ADD CONSTRAINT jobs\_job\_id\_pk

PRIMARY KEY (job\_id);

-- -------------------------------------------------------------------------

CREATE TABLE orders (

order\_id NUMERIC(6,0) NOT NULL,

order\_date DATE NOT NULL DEFAULT (CURRENT\_DATE),

shipped\_date DATE NOT NULL,

shipped\_status VARCHAR(15) NOT NULL,

customer\_id NUMERIC(6,0) NOT NULL,

employee\_id NUMERIC(6,0) NOT NULL,

return\_period VARCHAR(30) NOT NULL);

-- Primary Key

ALTER TABLE orders

ADD CONSTRAINT orders\_order\_id\_pk

PRIMARY KEY (order\_id);

-- Constraints

ALTER TABLE orders

ADD CONSTRAINT orders\_order\_date\_lt\_shipped\_date

CHECK (order\_date <= shipped\_date);

-- -------------------------------------------------------------------------

CREATE TABLE stores (

store\_id NUMERIC(6,0) NOT NULL,

store\_name VARCHAR(20) NOT NULL,

phone\_number VARCHAR(10) NOT NULL,

email VARCHAR(30) NOT NULL,

website VARCHAR(30),

address VARCHAR(50) NOT NULL,

zip\_code VARCHAR(10) NOT NULL,

manager\_id NUMERIC(6,0));

-- Primary Key

ALTER TABLE stores

ADD CONSTRAINT stores\_store\_id\_pk

PRIMARY KEY (store\_id);

-- Unique Keys

ALTER TABLE stores

ADD CONSTRAINT stores\_phone\_number\_uk

UNIQUE KEY (phone\_number);

ALTER TABLE stores

ADD CONSTRAINT stores\_email\_uk

UNIQUE KEY (email);

-- Constraints

ALTER TABLE stores

ADD CONSTRAINT stores\_email

CHECK (email LIKE '%@%.%');

-- -------------------------------------------------------------------------

CREATE TABLE suppliers (

supplier\_id NUMERIC(6,0) NOT NULL,

supplier\_name NUMERIC(6,0) NOT NULL,

supplier\_contact VARCHAR (10) NOT NULL,

supplier\_address VARCHAR (50) NOT NULL);

-- Primary Key

ALTER TABLE suppliers

ADD CONSTRAINT suppliers\_supplier\_id\_pk

PRIMARY KEY (supplier\_id);

-- -------------------------------------------------------------------------

CREATE TABLE warehouses (

warehouse\_id NUMERIC(6,0) NOT NULL,

warehouse\_name VARCHAR(30) NOT NULL,

address VARCHAR(50) NOT NULL,

capacity NUMERIC(8,2) NOT NULL,

zip\_code VARCHAR(10) NOT NULL);

-- Primary Key

ALTER TABLE warehouses

ADD CONSTRAINT warehouses\_warehouse\_id\_pk

PRIMARY KEY (warehouse\_id);

-- -------------------------------------------------------------------------

CREATE TABLE yarns (

yarn\_id NUMERIC(6,0) NOT NULL,

yarn\_type VARCHAR(15) NOT NULL,

yarn\_weight VARCHAR(15) NOT NULL,

yarn\_color VARCHAR(15) NOT NULL,

yarn\_price NUMERIC(9,2) NOT NULL,

yarn\_quality VARCHAR(20) NOT NULL,

yarn\_description VARCHAR(100));

-- Primary Key

ALTER TABLE yarns

ADD CONSTRAINT yarns\_yarn\_id\_pk

PRIMARY KEY (yarn\_id);

-- Constraints

ALTER TABLE yarns

ADD CONSTRAINT yarns\_\_yarn\_quality

CHECK (yarn\_quality IN ('common', 'rare', 'premium', 'custom'));

ALTER TABLE yarns

ADD CONSTRAINT yarns\_yarn\_price

CHECK (yarn\_price > 0);

-- -------------------------------------------------------------------------

CREATE TABLE zips (

zip\_code VARCHAR(10) NOT NULL,

city VARCHAR(20) NOT NULL,

province VARCHAR(20) NOT NULL,

country VARCHAR(20) NOT NULL);

-- Primary Key

ALTER TABLE zips

ADD CONSTRAINT zips\_zip\_code\_pk

PRIMARY KEY (zip\_code);

-- -------------------------------------------------------------------------

INSTERSECTION TABLES

-- -------------------------------------------------------------------------

CREATE TABLE inventories (

warehouse\_id NUMERIC(6,0) NOT NULL,

craft\_id NUMERIC(6,0) NOT NULL,

quantity NUMERIC(8,0) NOT NULL,

update\_date DATE NOT NULL DEFAULT (CURRENT\_DATE));

-- Primary Key

ALTER TABLE inventories

ADD CONSTRAINT inventories\_warehouse\_id\_craft\_id\_pk

PRIMARY KEY (warehouse\_id, craft\_id);

-- Foreign Key

ALTER TABLE inventories

ADD CONSTRAINT inventories\_warehouse\_id\_fk

FOREIGN KEY (warehouse\_id)

REFERENCES warehouses (warehouse\_id);

ALTER TABLE inventories

ADD CONSTRAINT inventories\_craft\_id\_fk

FOREIGN KEY (craft\_id)

REFERENCES crafts (craft\_id);

-- Constraints

ALTER TABLE inventories

ADD CONSTRAINT inventories\_quantity

CHECK (quantity > 0);

-- -------------------------------------------------------------------------

CREATE TABLE crafts\_orders (

craft\_id NUMERIC(6,0) NOT NULL,

order\_id NUMERIC(6,0) NOT NULL,

quantity NUMERIC(10,0) NOT NULL);

-- Primary Key

ALTER TABLE crafts\_orders

ADD CONSTRAINT crafts\_orders\_craft\_id\_order\_id\_pk

PRIMARY KEY (craft\_id, order\_id);

-- Foreign Key

ALTER TABLE crafts\_orders

ADD CONSTRAINT crafts\_orders\_craft\_id\_fk

FOREIGN KEY (craft\_id)

REFERENCES crafts (craft\_id);

ALTER TABLE crafts\_orders

ADD CONSTRAINT crafts\_orders\_order\_id\_fk

FOREIGN KEY (order\_id)

REFERENCES orders (order\_id);

-- -------------------------------------------------------------------------

CREATE TABLE crafts\_yarns (

craft\_id NUMERIC(6,0) NOT NULL,

yarn\_id NUMERIC(6,0) NOT NULL,

quantity NUMERIC(6,0));

-- Primary Keys

ALTER TABLE crafts\_yarns

ADD CONSTRAINT crafts\_yarns\_craft\_id\_yarn\_id\_pk

PRIMARY KEY (craft\_id, yarn\_id);

-- Foreign Keys

ALTER TABLE crafts\_yarns

ADD CONSTRAINT crafts\_yarns\_craft\_id\_fk

FOREIGN KEY (craft\_id)

REFERENCES crafts (craft\_id);

ALTER TABLE crafts\_yarns

ADD CONSTRAINT crafts\_yarns\_yarn\_id\_fk

FOREIGN KEY (yarn\_id)

REFERENCES yarns (yarn\_id);

-- Constraints

ALTER TABLE crafts\_yarns

ADD CONSTRAINT crafts\_yarns\_quantity

CHECK (quantity > 0);

-- -------------------------------------------------------------------------

CREATE TABLE warehouses\_stores (

warehouse\_id NUMERIC(6,0) NOT NULL,

store\_id NUMERIC(6,0) NOT NULL);

-- Primary Keys

ALTER TABLE warehouses\_stores

ADD CONSTRAINT warehouses\_stores\_warehouse\_id\_store\_id\_pk

PRIMARY KEY (warehouse\_id, store\_id);

-- Foreign Keys

ALTER TABLE warehouses\_stores

ADD CONSTRAINT warehouses\_stores\_warehouse\_id\_fk

FOREIGN KEY (warehouse\_id)

REFERENCES warehouses (warehouse\_id);

ALTER TABLE warehouses\_stores

ADD CONSTRAINT warehouses\_stores\_store\_id\_fk

FOREIGN KEY (store\_id)

REFERENCES stores (store\_id);

-- -------------------------------------------------------------------------

CREATE TABLE yarns\_suppliers (

yarn\_id NUMERIC(6,0) NOT NULL,

supplier\_id NUMERIC(6,0) NOT NULL);

-- Primary Key

ALTER TABLE yarns\_suppliers

ADD CONSTRAINT yarns\_suppliers\_yarn\_id\_supplier\_id\_pk

PRIMARY KEY (yarn\_id, supplier\_id);

-- Foreign Key

ALTER TABLE yarns\_suppliers

ADD CONSTRAINT yarns\_suppliers\_yarn\_id\_fk

FOREIGN KEY (yarn\_id)

REFERENCES yarns (yarn\_id);

ALTER TABLE yarns\_suppliers

ADD CONSTRAINT yarns\_suppliers\_supplier\_id\_fk

FOREIGN KEY (supplier\_id)

REFERENCES suppliers (supplier\_id);

-- -------------------------------------------------------------------------

FOREIGN KEYS

-- -------------------------------------------------------------------------

CUSTOMERS

-- FOREIGN Keys

ALTER TABLE customers

ADD CONSTRAINT customers\_favorite\_craft\_fk

FOREIGN KEY (favorite\_craft)

REFERENCES crafts (craft\_id);

ALTER TABLE customers

ADD CONSTRAINT customers\_zip\_code\_fk

FOREIGN KEY (zip\_code)

REFERENCES zips (zip\_code);

ALTER TABLE customers

ADD CONSTRAINT customers\_preferred\_sales\_rep\_fk

FOREIGN KEY (preferred\_sales\_rep)

REFERENCES employees (employee\_id);

-- -------------------------------------------------------------------------

DEPARTMENTS

-- Foreign Key

ALTER TABLE departments

ADD CONSTRAINT departments\_manager\_id\_fk

FOREIGN KEY (manager\_id)

REFERENCES employees (employee\_id);

-- -------------------------------------------------------------------------

EMPLOYEES

-- Foreign Key

ALTER TABLE employees

ADD CONSTRAINT employees\_manager\_id\_fk

FOREIGN KEY (manager\_id)

REFERENCES employees (employee\_id);

ALTER TABLE employees

ADD CONSTRAINT employees\_store\_id\_fk

FOREIGN KEY (store\_id)

REFERENCES stores (store\_id);

ALTER TABLE employees

ADD CONSTRAINT employees\_department\_id\_fk

FOREIGN KEY (department\_id)

REFERENCES departments (department\_id);

ALTER TABLE employees

ADD CONSTRAINT employees\_job\_id\_fk

FOREIGN KEY (job\_id)

REFERENCES jobs (job\_id);

-- -------------------------------------------------------------------------

ORDERS

-- Foreign Key

ALTER TABLE orders

ADD CONSTRAINT orders\_customer\_id\_fk

FOREIGN KEY (customer\_id)

REFERENCES customers (customer\_id);

ALTER TABLE orders

ADD CONSTRAINT orders\_employee\_id\_fk

FOREIGN KEY (employee\_id)

REFERENCES employees (employee\_id);

-- -------------------------------------------------------------------------

STORES

-- Foreign Key

ALTER TABLE stores

ADD CONSTRAINT stores\_zip\_code\_fk

FOREIGN KEY (zip\_code)

REFERENCES zips (zip\_code);

ALTER TABLE stores

ADD CONSTRAINT stores\_manager\_id\_fk

FOREIGN KEY (manager\_id)

REFERENCES employees (manager\_id);

-- -------------------------------------------------------------------------

WAREHOUSES

-- Foreign Key

ALTER TABLE warehouses

ADD CONSTRAINT warehouses\_zip\_code\_fk

FOREIGN KEY (zip\_code)

REFERENCES zips (zip\_code);

**Test Cases**

**CUSTOMERS Table**

-- ---------------------------------------------------------------------------

customers\_customer\_id\_pk

-- ---------------------------------------------------------------------------

-- \*\* Valid test \*\*

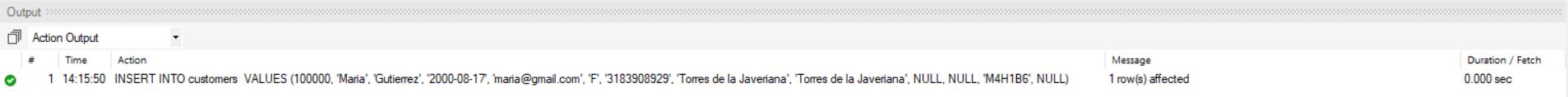
-- Expected results: Insert statement add a row in the customers table

-- Action:

INSERT INTO customers

VALUES (100000, 'Maria', 'Gutierrez', '2000-08-17', 'maria@gmail.com', 'F', '3183908929', 'Torres de la Javeriana', 'Torres de la Javeriana', NULL, NULL, 'M4H1B6', NULL);

-- Test results: 1 row(s) affected



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Insert statement fails to add a row in the customers table

-- Action:

INSERT INTO customers

VALUES (100000, 'John', 'Alexander', '1995-10-17', 'alexander@gmail.com', 'M', '6478601662', '347 Lawson Road', '347 Lawson Road', NULL, NULL, 'M1C2J8', NULL);

-- Test results: Error Code: 1062. Duplicate entry '100000' for key 'customers.PRIMARY'



-- ---------------------------------------------------------------------------

customers\_email\_uk

-- ---------------------------------------------------------------------------

\*\* Valid test \*\*

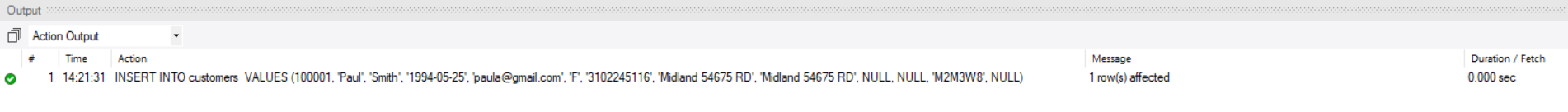
-- Expected results: Insert statement add a row in the customers table

-- Action:

INSERT INTO customers

VALUES (100001, 'Paul', 'Smith', '1994-05-25', 'paula@gmail.com', 'F', '3102245116', 'Midland 54675 RD', 'Midland 54675 RD', NULL, NULL, 'M2M3W8', NULL);

-- Test results: 1 row(s) affected



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the customers table

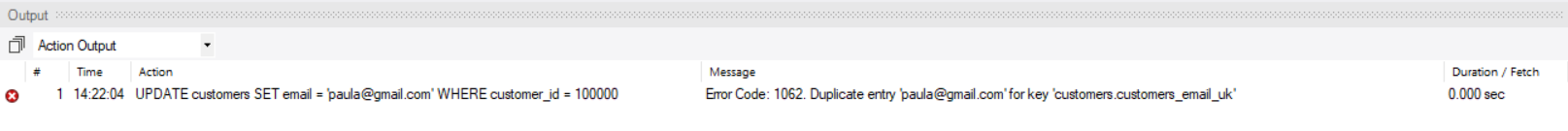
-- Action:

UPDATE customers

SET email = 'paula@gmail.com'

WHERE customer\_id = 100000;

-- Test results: Error Code: 1062. Duplicate entry 'paula@gmail.com' for key 'customers.customers\_email\_uk'



-- ---------------------------------------------------------------------------

customers\_contact\_number\_uk

-- ---------------------------------------------------------------------------

-- \*\* Valid test \*\*

-- Expected results: Update statement modifies a row in the customers table

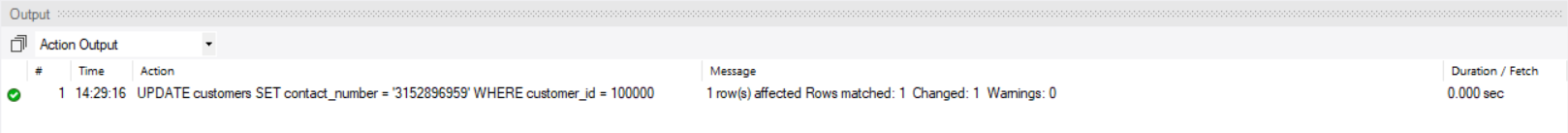
-- Action:

UPDATE customers

SET contact\_number = '3152896959'

WHERE customer\_id = 100000;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the customers table

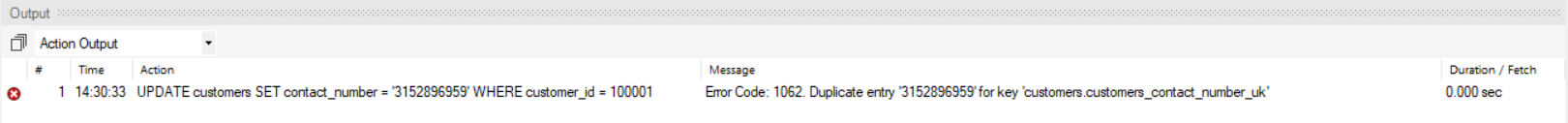
-- Action:

UPDATE customers

SET contact\_number = '3152896959'

WHERE customer\_id = 100001;

-- Test results: Error Code: 1062. Duplicate entry '3152896959' for key 'customers.customers\_contact\_number\_uk'



-- ---------------------------------------------------------------------------

customers\_favorite\_craft\_fk

-- ---------------------------------------------------------------------------

-- IMPORTANT: 000001, 000002, 000003 craft\_id exits in crafts table

-- \*\* Valid test \*\*

-- Expected results: Update statement modifies a row in the customers table

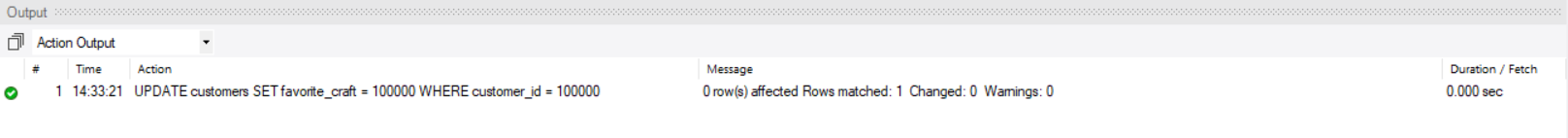
-- Action:

UPDATE customers

SET favorite\_craft = 100000

WHERE customer\_id = 100000;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the customers table

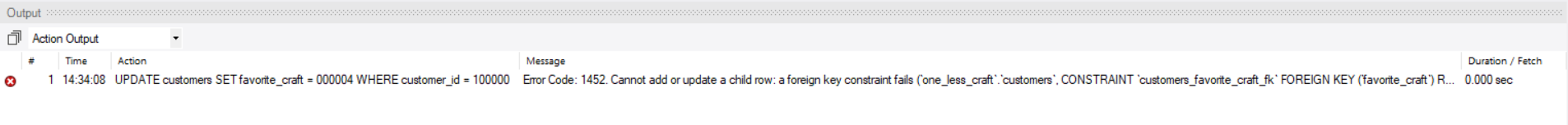
-- Action:

UPDATE customers

SET favorite\_craft = 000004

WHERE customer\_id = 100000;

-- Test results: Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (`one\_less\_craft`.`customers`, CONSTRAINT `customers\_favorite\_craft\_fk` FOREIGN KEY (`favorite\_craft`) REFERENCES `crafts` (`craft\_id`))



-- ---------------------------------------------------------------------------

customers\_zip\_code\_fk

-- ---------------------------------------------------------------------------

-- IMPORTANT: M1C2J8, M4H1B6, M2M3W8 zip\_code exits in zips table

-- \*\* Valid test \*\*

-- Expected results: Update statement modifies a row in the customers table

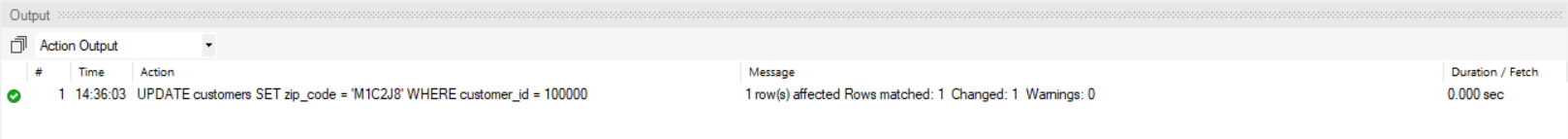
-- Action:

UPDATE customers

SET zip\_code = 'M1C2J8'

WHERE customer\_id = 100000;

-- Test results: 1 row Inserted



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the customers table

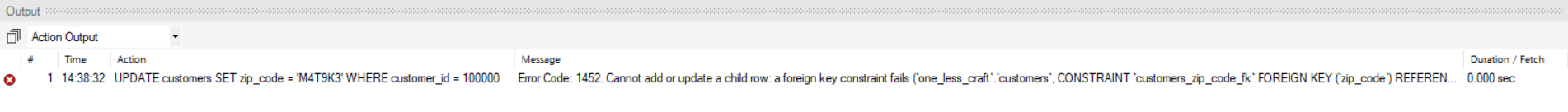
-- Action:

UPDATE customers

SET zip\_code = 'M4T9K3'

WHERE customer\_id = 100000;

-- Test results: Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (`one\_less\_craft`.`customers`, CONSTRAINT `customers\_zip\_code\_fk` FOREIGN KEY (`zip\_code`) REFERENCES `zips` (`zip\_code`))



-- ---------------------------------------------------------------------------

customers\_preferred\_sales\_rep\_fk

-- ---------------------------------------------------------------------------

-- IMPORTANT: 100001, 100002, 100003 employee\_id exits in employees table

-- \*\* Valid test \*\*

-- Expected results: Update statement modifies a row in the customers table

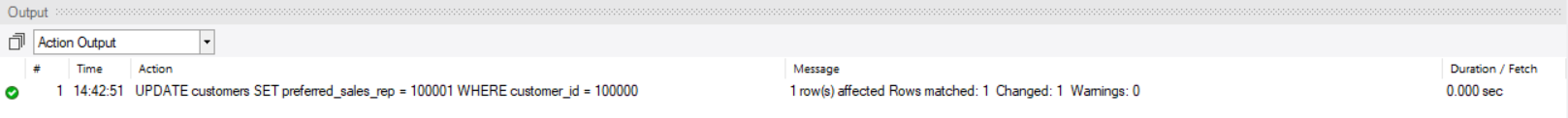
-- Action:

UPDATE customers

SET preferred\_sales\_rep = 100001

WHERE customer\_id = 100000;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the customers table

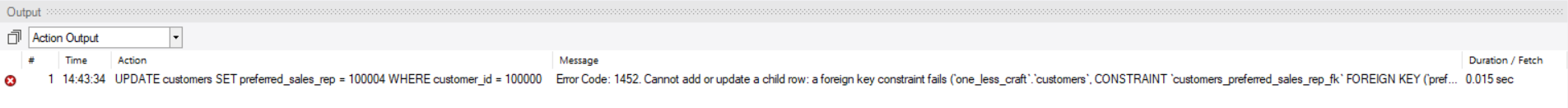
-- Action:

UPDATE customers

SET preferred\_sales\_rep = 100004

WHERE customer\_id = 100000;

-- Test results: Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (`one\_less\_craft`.`customers`, CONSTRAINT `customers\_preferred\_sales\_rep\_fk` FOREIGN KEY (`preferred\_sales\_rep`) REFERENCES `employees` (`employee\_id`))



-- ---------------------------------------------------------------------------

customers\_email

-- ---------------------------------------------------------------------------

-- \*\* Valid test \*\*

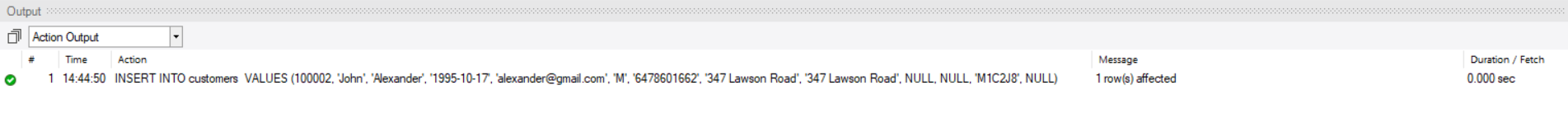
-- Expected results: Insert statement add a row in the customers table

-- Action:

INSERT INTO customers

VALUES (100002, 'John', 'Alexander', '1995-10-17', 'alexander@gmail.com', 'M', '6478601662', '347 Lawson Road', '347 Lawson Road', NULL, NULL, 'M1C2J8', NULL);

-- Test results: 1 row(s) affected



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the customers table

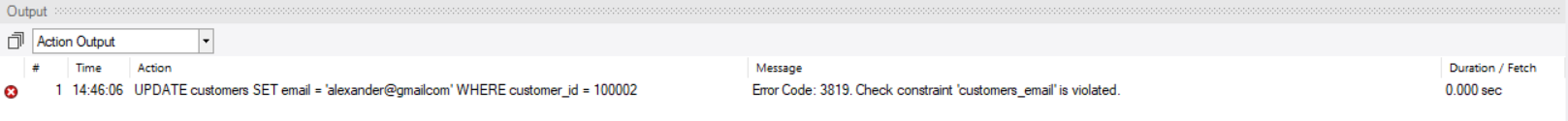
-- Action:

UPDATE customers

SET email = 'alexander@gmailcom'

WHERE customer\_id = 100002;

-- Test results: Error Code: 3819. Check constraint 'customers\_email' is violated.



-- ---------------------------------------------------------------------------

customers\_gender

-- ---------------------------------------------------------------------------

-- \*\* Valid test \*\*

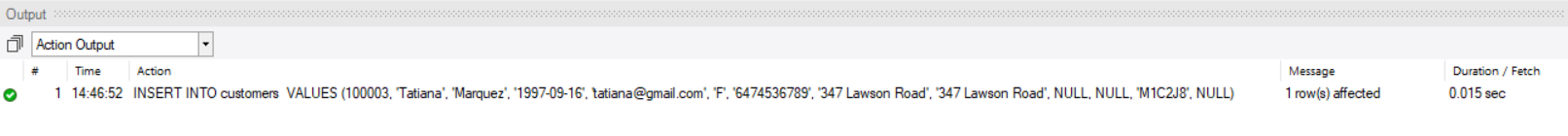
-- Expected results: Insert statement add a row in the customers table

-- Action:

INSERT INTO customers

VALUES (100003, 'Tatiana', 'Marquez', '1997-09-16', 'tatiana@gmail.com', 'F', '6474536789', '347 Lawson Road', '347 Lawson Road', NULL, NULL, 'M1C2J8', NULL);

-- Test results: 1 row(s) affected



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the customers table

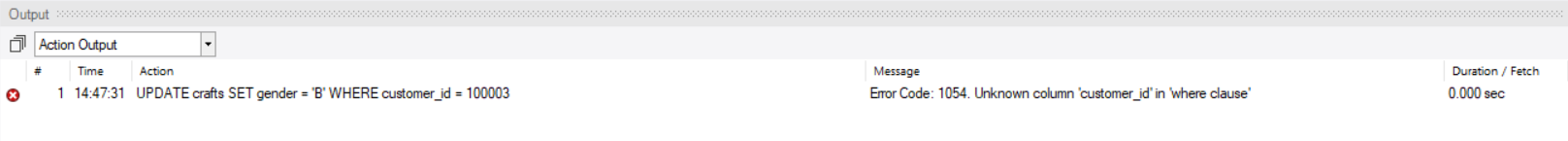
-- Action:

UPDATE crafts

SET gender = 'B'

WHERE customer\_id = 100003;

-- Test results: Error Code: 1054. Unknown column 'customer\_id' in 'where clause'



**EMPLOYEES Table**

-- ---------------------------------------------------------------------------

employees\_employee\_id\_pk

-- ---------------------------------------------------------------------------

\*\* Valid test \*\*

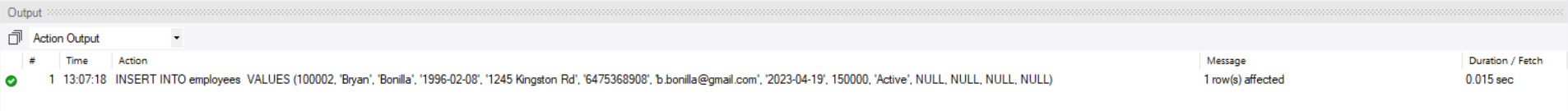
-- Expected results: Insert statement add a row in the employees table

-- Action:

INSERT INTO employees

VALUES (100002, 'Bryan', 'Bonilla', '1996-02-08', '1245 Kingston Rd', '6475368908', 'b.bonilla@gmail.com', '2023-04-19', 150000, 'Active', NULL, NULL, NULL, NULL);

-- Test results: 1 row(s) affected



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

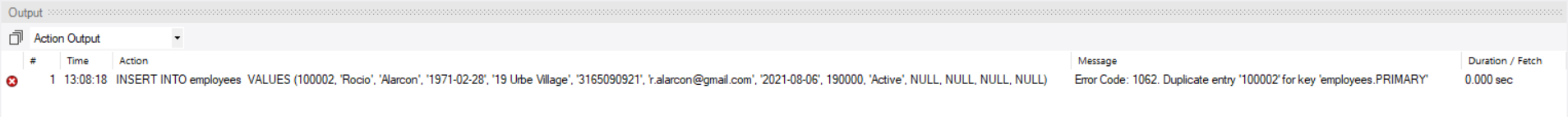
-- Expected results: Insert statement fails to add a row in the employees table

-- Action:

INSERT INTO employees

VALUES (100002, 'Rocio', 'Alarcon', '1971-02-28', '19 Urbe Village', '3165090921', 'r.alarcon@gmail.com', '2021-08-06', 190000, 'Active', NULL, NULL, NULL, NULL);

-- Test results: Error Code: 1062. Duplicate entry '100002' for key 'employees.PRIMARY'



-- ---------------------------------------------------------------------------

employees\_contact\_number\_uk

-- ---------------------------------------------------------------------------

\*\* Valid test \*\*

-- Expected results: Update statement modifies a row in the employees table

-- Action:

INSERT INTO employees

VALUES (100003, 'Rocio', 'Alarcon', '1971-02-28', '19 Urbe Village', '3165090921', 'r.alarcon@gmail.com', '2021-08-06', 190000, 'Active', NULL, NULL, NULL, NULL);

-- Test results: 1 row(s) affected



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the employees table

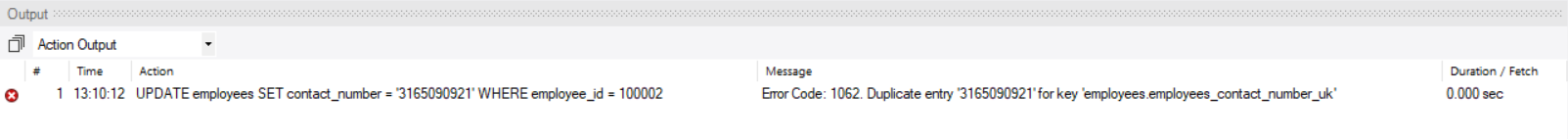
-- Action:

UPDATE employees

SET contact\_number = '3165090921'

WHERE employee\_id = 100002;

-- Test results: Error Code: 1062. Duplicate entry '3165090921' for key 'employees.employees\_contact\_number\_uk'



-- ---------------------------------------------------------------------------

employees\_email\_uk

-- ---------------------------------------------------------------------------

\*\* Valid test \*\*

-- Expected results: Insert statement add a row in the employees table

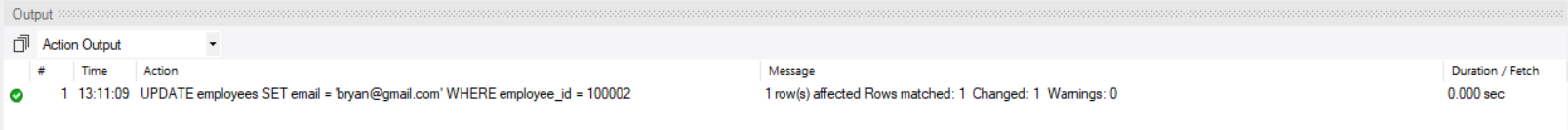
-- Action:

UPDATE employees

SET email = 'bryan@gmail.com'

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the employees table

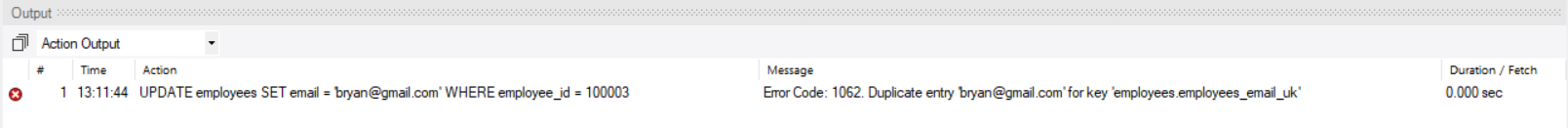
-- Action:

UPDATE employees

SET email = 'bryan@gmail.com'

WHERE employee\_id = 100003;

-- Test results: Error Code: 1062. Duplicate entry 'bryan@gmail.com' for key 'employees.employees\_email\_uk'



-- ---------------------------------------------------------------------------

employees\_manager\_id\_fk

-- ---------------------------------------------------------------------------

\*\* Valid test \*\*

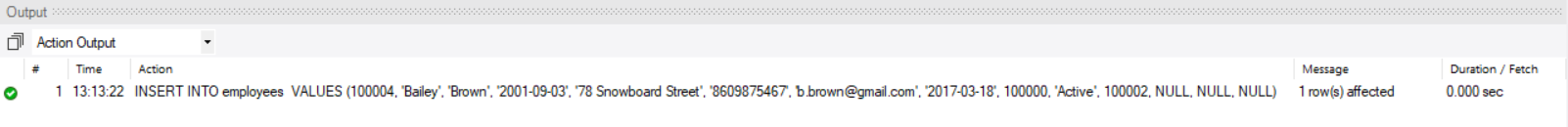
-- Expected results: Insert statement add a row in the employees table

-- Action:

INSERT INTO employees

VALUES (100004, 'Bailey', 'Brown', '2001-09-03', '78 Snowboard Street', '8609875467', 'b.brown@gmail.com', '2017-03-18', 100000, 'Active', 100002, NULL, NULL, NULL);

-- Test results: 1 row(s) affected



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the employees table

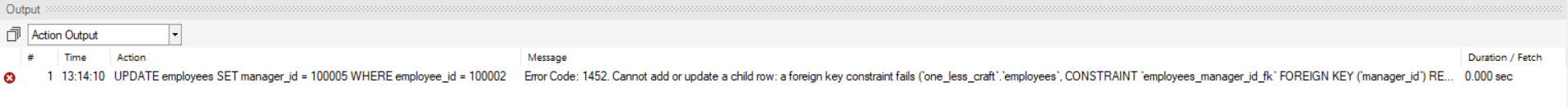
-- Action:

UPDATE employees

SET manager\_id = 100005

WHERE employee\_id = 100002;

-- Test results: Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (`one\_less\_craft`.`employees`, CONSTRAINT `employees\_manager\_id\_fk` FOREIGN KEY (`manager\_id`) REFERENCES `employees` (`employee\_id`))



-- ---------------------------------------------------------------------------

employees\_store\_id\_fk

-- ---------------------------------------------------------------------------

-- IMPORTANT: 123456, 123457, 123458 store\_id exist in the stores table.

-- \*\* Valid test \*\*

-- Expected results: Update statement modifies a row in the employees table

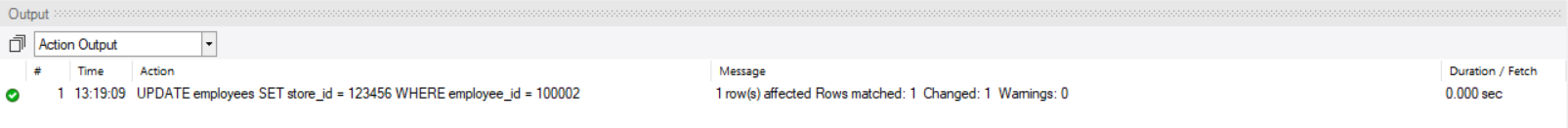
-- Action:

UPDATE employees

SET store\_id = 123456

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the employees table

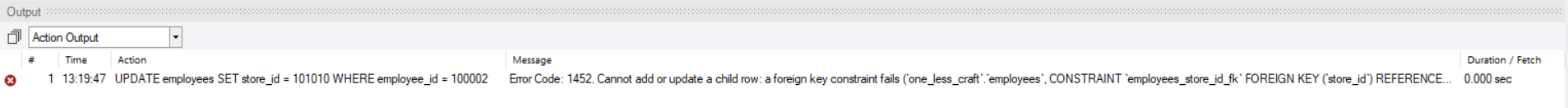
-- Action:

UPDATE employees

SET store\_id = 101010

WHERE employee\_id = 100002;

-- Test results: Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (`one\_less\_craft`.`employees`, CONSTRAINT `employees\_store\_id\_fk` FOREIGN KEY (`store\_id`) REFERENCES `stores` (`store\_id`))



-- ---------------------------------------------------------------------------

employees\_department\_id\_fk

-- ---------------------------------------------------------------------------

-- IMPORTANT: 000111, 000222, 000333 department\_id exist in the departments table.

-- \*\* Valid test \*\*

-- Expected results: Update statement modifies a row in the employees table

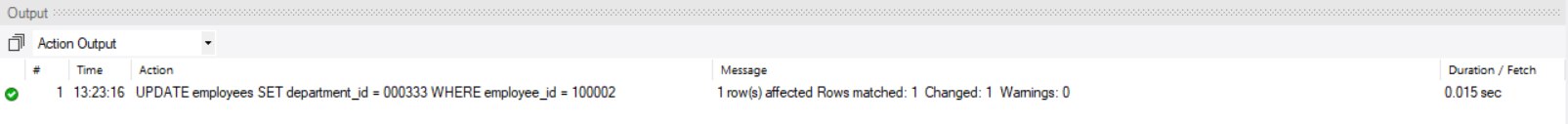
-- Action:

UPDATE employees

SET department\_id = 000333

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the employees table

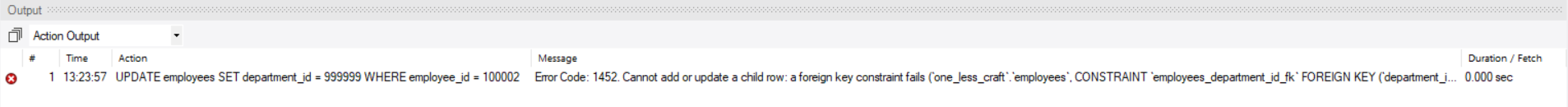
-- Action:

UPDATE employees

SET department\_id = 999999

WHERE employee\_id = 100002;

-- Test results: Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (`one\_less\_craft`.`employees`, CONSTRAINT `employees\_department\_id\_fk` FOREIGN KEY (`department\_id`) REFERENCES `departments` (`department\_id`))



-- ---------------------------------------------------------------------------

employees\_job\_id\_fk

-- ---------------------------------------------------------------------------

-- IMPORTANT: 111000, 222000, 333000 job\_id exist in the jobs table.

-- \*\* Valid test \*\*

-- Expected results: Update statement modifies a row in the employees table

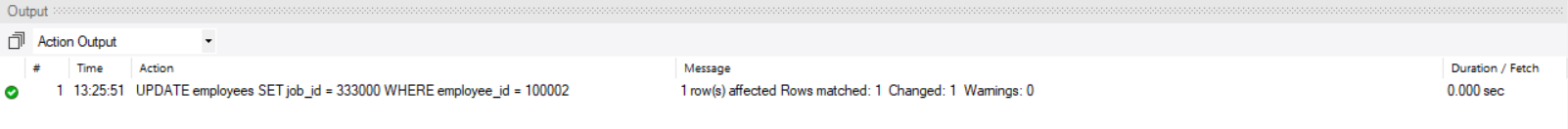
-- Action:

UPDATE employees

SET job\_id = 333000

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the employees table

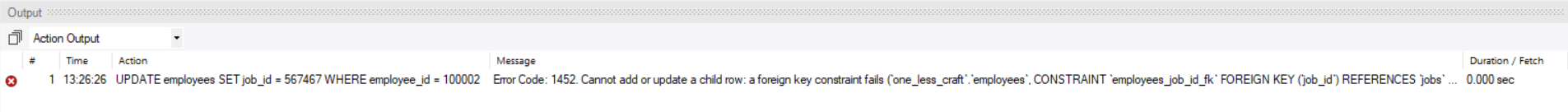
-- Action:

UPDATE employees

SET job\_id = 567467

WHERE employee\_id = 100002;

-- Test results: Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (`one\_less\_craft`.`employees`, CONSTRAINT `employees\_job\_id\_fk` FOREIGN KEY (`job\_id`) REFERENCES `jobs` (`job\_id`))



-- ---------------------------------------------------------------------------

employees\_salary\_range

-- ---------------------------------------------------------------------------

\*\* Valid test #1 \*\*

-- Expected results: Update statement modifies a row in the employees table

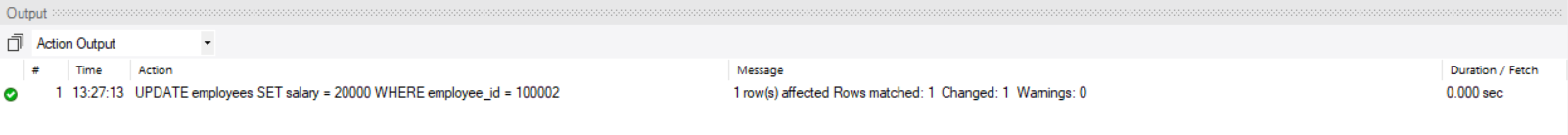
-- Action:

UPDATE employees

SET salary = 20000

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



\*\* Valid test #2 \*\*

-- Expected results: Update statement modifies a row in the employees table

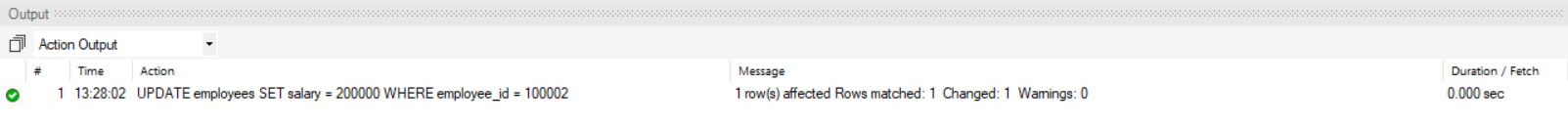
-- Action:

UPDATE employees

SET salary = 200000

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



\*\* Valid test #3 \*\*

-- Expected results: Update statement modifies a row in the employees table

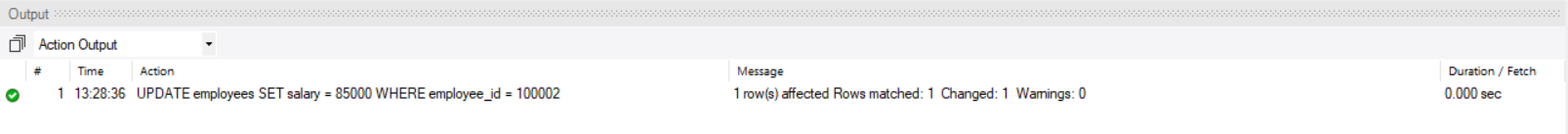
-- Action:

UPDATE employees

SET salary = 85000

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



-- ---------------------------------------------------------------------------

\*\* InValid test #1 \*\*

-- Expected results: Update statement fails to modify a row in the employees table

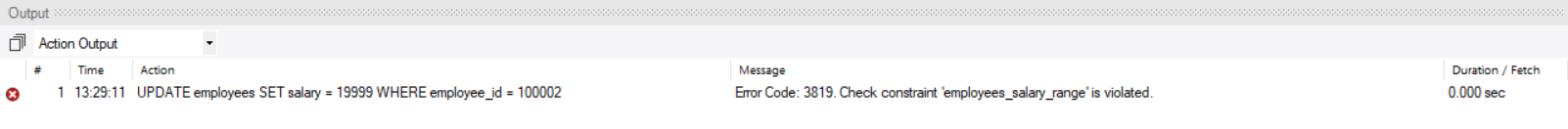
-- Action:

UPDATE employees

SET salary = 19999

WHERE employee\_id = 100002;

-- Test results: Error Code: 3819. Check constraint 'employees\_salary\_range' is violated.



\*\* InValid test #2 \*\*

-- Expected results: Update statement fails to modify a row in the employees table

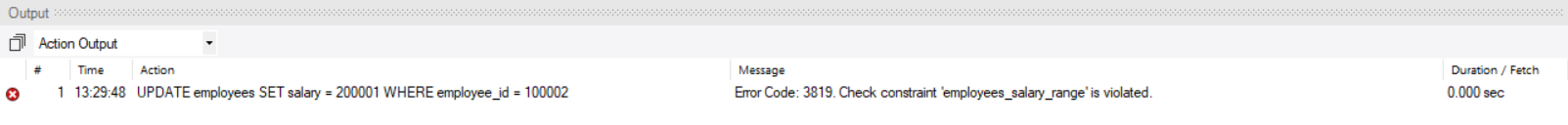
-- Action:

UPDATE employees

SET salary = 200001

WHERE employee\_id = 100002;

-- Test results: Error Code: 3819. Check constraint 'employees\_salary\_range' is violated.



-- ---------------------------------------------------------------------------

employees\_emp\_status

-- ---------------------------------------------------------------------------

\*\* Valid test #1 \*\*

-- Expected results: Update statement modifies a row in the employees table

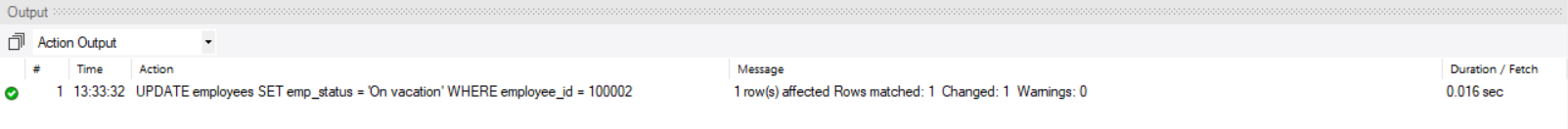
-- Action:

UPDATE employees

SET emp\_status = 'On vacation'

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



\*\* Valid test #2 \*\*

-- Expected results: Update statement modifies a row in the employees table

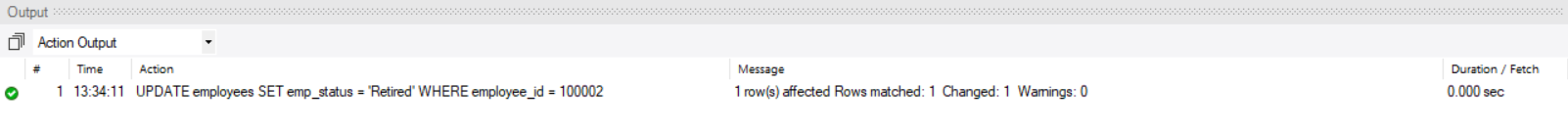
-- Action:

UPDATE employees

SET emp\_status = 'Retired'

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



\*\* Valid test #3 \*\*

-- Expected results: Update statement modifies a row in the employees table

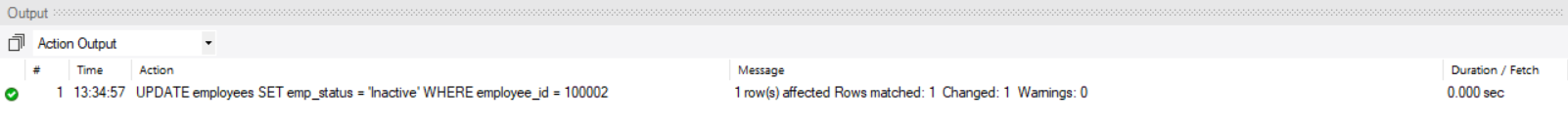
-- Action:

UPDATE employees

SET emp\_status = 'Inactive'

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



\*\* Valid test #4 \*\*

-- Expected results: Update statement modifies a row in the employees table

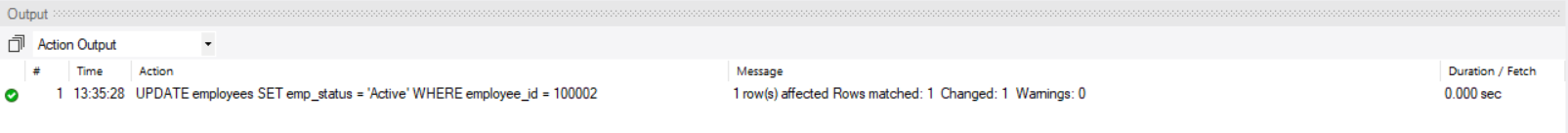
-- Action:

UPDATE employees

SET emp\_status = 'Active'

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the employees table

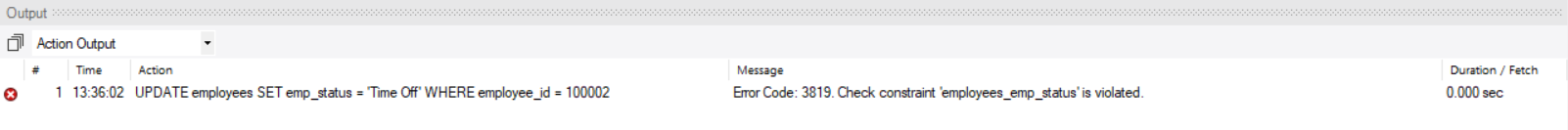
-- Action:

UPDATE employees

SET emp\_status = 'Time Off'

WHERE employee\_id = 100002;

-- Test results:



-- ---------------------------------------------------------------------------

employees\_hire\_date\_gt\_date\_of\_birth

-- ---------------------------------------------------------------------------

-- \*\* Valid test \*\*

-- Expected results: Update statement modifies a row in the employees table

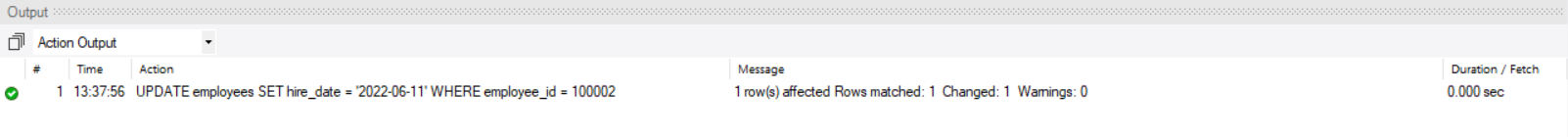
-- Action:

UPDATE employees

SET hire\_date = '2022-06-11' -- DOB = '1996-02-08'

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



-- ---------------------------------------------------------------------------

\*\* InValid test \*\*

-- Expected results: Update statement fails to modify a row in the employees table

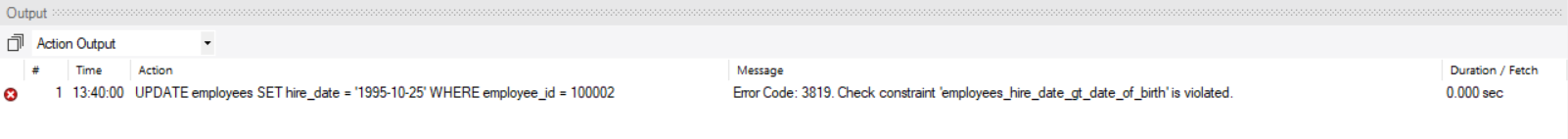
-- Action:

UPDATE employees

SET hire\_date = '1995-10-25'

WHERE employee\_id = 100002;

-- Test results: Error Code: 3819. Check constraint 'employees\_hire\_date\_gt\_date\_of\_birth' is violated.



-- ---------------------------------------------------------------------------

employees\_email

-- ---------------------------------------------------------------------------

\*\* Valid test \*\*

-- Expected results: Update statement modifies a row in the employees table

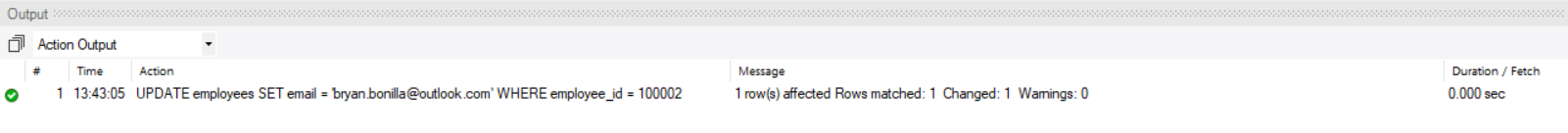
-- Action:

UPDATE employees

SET email = 'bryan.bonilla@outlook.com'

WHERE employee\_id = 100002;

-- Test results: 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0



-- ---------------------------------------------------------------------------

\*\* InValid test #1 \*\*

-- Expected results: Update statement fails to modify a row in the employees table

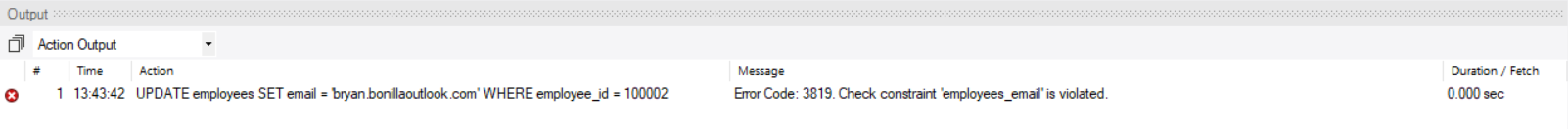
-- Action:

UPDATE employees

SET email = 'bryan.bonillaoutlook.com'

WHERE employee\_id = 100002;

-- Test results: Error Code: 3819. Check constraint 'employees\_email' is violated.



\*\* InValid test #2 \*\*

-- Expected results: Update statement fails to modify a row in the employees table

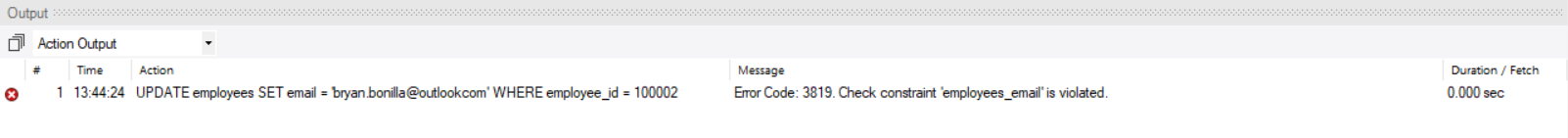
-- Action:

UPDATE employees

SET email = 'bryan.bonilla@outlookcom'

WHERE employee\_id = 100002;

-- Test results: Error Code: 3819. Check constraint 'employees\_email' is violated.

****