

```

/*****
* NAME: Samuel Sovi
* CLASS: CPSC 321 Section 1
* DATE: 9/22/22
* HOMEWORK: hw1
* DESCRIPTION: Implementing Tables and Schemas to practice creating
* relationships between tables using foreign keys
*
* ADDITIONAL NOTE:
*     Useful commands
*     1) mysql -p -h cps-database
*     2) use ssovi_DB;
*     3) show tables;
*     4) exit
*     5) source hw1-schema.sql
*****/

```

40/40

```
-- TODO: add drop table statements
```

```
DROP TABLE IF EXISTS trip;
DROP TABLE IF EXISTS vehicle;
DROP TABLE IF EXISTS default_plan;
DROP TABLE IF EXISTS allowed_plan;
DROP TABLE IF EXISTS vehicle_type;
DROP TABLE IF EXISTS price_plan;
DROP TABLE IF EXISTS customer;
```

```
-- TODO: add create table statements
```

```
-- customer table used to store customer info
```

```
CREATE TABLE customer (
    -- int refers to unique customer id
    c_id SMALLINT UNSIGNED NOT NULL,
    -- first name of customer
    first_name VARCHAR(20) NOT NULL,
    -- last name of customer
    last_name VARCHAR(20) NOT NULL,
    -- character string of user's email
    email VARCHAR(50) NOT NULL,
    PRIMARY KEY (c_id)
);
```

```
-- price_plan table used to store different price plans
```

```
CREATE TABLE price_plan (
    -- character limit since primary key
    plan_name VARCHAR(50) NOT NULL,
    -- int refers to number of cents
    price_per_min SMALLINT NOT NULL,
    -- int refers to number of cents
    unlock_price MEDIUMINT NOT NULL,
    -- refers to number of minutes since start
    rate_start_min MEDIUMINT NOT NULL,
    PRIMARY KEY (plan_name)
);
```

```
-- vehicle_type table used to store different vehicle types
```

```
CREATE TABLE vehicle_type (
    -- unique id for each vehicle type
```

```

vt_id SMALLINT UNSIGNED NOT NULL,
-- form_factor is either e-scooter or e-bike
form_factor VARCHAR(10) NOT NULL,
-- make of the vehicle (e.g. "Segway")
make VARCHAR(20) NOT NULL,
-- model of the vehicle (e.g. "Ninebot")
model VARCHAR(20) NOT NULL,
-- trim level of the vehicle (e.g. "ES4")
trim_level VARCHAR(20) NOT NULL,
-- alternate name of the vehicle (e.g. "Lime-S ES4")
alt_name VARCHAR(50),
-- max speed of the vehicle in mph (e.g. 80.3)
max_spd FLOAT(4,1),
-- max range on a full charge in miles (e.g. 26.2)
max_range FLOAT(5,1) NOT NULL,
-- weight of vehicle in pounds (e.g. 60.3)
weight SMALLINT UNSIGNED,
PRIMARY KEY (vt_id),
-- form factor must be "e-scooter" or "e-bike"
CONSTRAINT valid_form_factor CHECK(form_factor = "e-scooter" OR form_factor
= "e-bike")
);

```

- ENUM

```

CREATE TABLE allowed_plan(
-- unique id for each vehicle type (from vehicle type)
vt_id SMALLINT UNSIGNED NOT NULL,
-- character limit since primary key (uses plan_name from price_plan)
price_plan VARCHAR(50) NOT NULL,
PRIMARY KEY (vt_id, price_plan),
FOREIGN KEY (vt_id) REFERENCES vehicle_type (vt_id),
FOREIGN KEY (price_plan) REFERENCES price_plan (plan_name)
);

```

```

CREATE TABLE default_plan(
-- unique id for each vehicle type (from vehicle type)
vt_id SMALLINT UNSIGNED NOT NULL,
-- character limit since primary key (uses plan_name from price_plan)
price_plan VARCHAR(50) NOT NULL,
PRIMARY KEY (vt_id),
FOREIGN KEY (vt_id, price_plan) REFERENCES allowed_plan (vt_id, price_plan)
-- FOREIGN KEY (price_plan) REFERENCES allowed_plan (price_plan)
);

```

✓

```

CREATE TABLE vehicle(
-- each vehicle is assigned a vehicle id
v_id SMALLINT UNSIGNED NOT NULL,
-- unique id for each vehicle type (from vehicle type)
vt_id SMALLINT UNSIGNED NOT NULL,
-- (boolean) whether or not vehicle is in circulation
in_circ BOOLEAN NOT NULL,
-- (boolean) whether or not vehicle is reserved
is_reserved BOOLEAN NOT NULL,
-- (boolean) whether or not vehicle is disabled
is_disabled BOOLEAN NOT NULL,
-- latitude of the vehicle with domain: (-90.000000, 90.000000)
lat DOUBLE(10,6) SIGNED NOT NULL,
-- longitude of the vehicle with domain: (-180.000000, 180.000000)
lon DOUBLE(10,6) SIGNED NOT NULL,
-- current fuel percentage with domain: (0.0000, 100.00)
curr_fuel_pct FLOAT(5,2) NOT NULL,

```

```

-- current range of the vehicle left (in miles)
cur_range FLOAT(5,1) NOT NULL,
PRIMARY KEY (v_id),
FOREIGN KEY (vt_id) REFERENCES vehicle_type (vt_id),
CONSTRAINT valid_curr_fuel_pct CHECK(curr_fuel_pct <= 100 AND curr_fuel_pct
>= 0),
CONSTRAINT valid_lat CHECK(lat <= 90 AND lat >= -90),
CONSTRAINT valid_lon CHECK(lon <= 180 AND lon >= -180)
);

CREATE TABLE trip(
-- each trip is assigned a unique id
t_id SMALLINT UNSIGNED NOT NULL,
-- int refers to unique customer id
c_id SMALLINT UNSIGNED NOT NULL,
-- each vehicle is assigned a vehicle id
v_id SMALLINT UNSIGNED NOT NULL,
-- character limit since primary key (usues plan_name from price_plan)
price_plan VARCHAR(50) NOT NULL,
-- start datetime in format YYYY-MM-DD HH:MM:SS
start_dtime DATETIME,
-- end datetime in format YYYY-MM-DD HH:MM:SS
end_dtime DATETIME,
-- starting latitude of the trip with domain: (-90.000000, 90.000000)
start_lat DOUBLE(10,6) SIGNED NOT NULL,
-- starting longitude of the trip with domain: (-180.000000, 180.000000)
start_lon DOUBLE(10,6) SIGNED NOT NULL,
-- ending latitude of the trip with domain: (-90.000000, 90.000000)
end_lat DOUBLE(10,6) SIGNED NOT NULL,
-- ending longitude of the trip with domain: (-180.000000, 180.000000)
end_lon DOUBLE(10,6) SIGNED NOT NULL,
PRIMARY KEY (t_id),
FOREIGN KEY (c_id) REFERENCES customer (c_id),
FOREIGN KEY (v_id) REFERENCES vehicle (v_id),
FOREIGN KEY (price_plan) REFERENCES price_plan (plan_name),
CONSTRAINT valid_start_lat CHECK(start_lat <= 90 AND start_lat >= -90),
CONSTRAINT valid_start_lon CHECK(start_lon <= 180 AND start_lon >= -180),
CONSTRAINT valid_end_lat CHECK(end_lat <= 90 AND end_lat >= -90),
CONSTRAINT valid_end_lon CHECK(end_lon <= 180 AND end_lon >= -180),
CONSTRAINT valid_end_dtime CHECK(end_dtime > start_dtime)
);

```

```

/*****
* NAME: Samuel Sovi
* CLASS: CPSC 321 Section 1
* DATE: 9/22/22
* HOMEWORK: hw1
* DESCRIPTION: Implementing Tables and Schemas to practice creating
* relationships between tables using foreign keys
*
* ADDITIONAL NOTE:
*     Useful commands
*     1) mysql -p -h cps-database
*     2) use ssovi_DB;
*     3) show tables;
*     4) exit
*****/

-- TODO: add insert statements

INSERT INTO customer VALUES (1, "Samuel", "Sovi", "samsovi123@gmail.com");
INSERT INTO customer VALUES (2, "Daniel", "Rodricks", "danrodricks123@gmail.com"
);

INSERT INTO price_plan VALUES ("standard plan", 2, 10, 3);
INSERT INTO price_plan VALUES ("plan premium plus", 5, 20, 1);

INSERT INTO vehicle_type VALUES (12, "e-scooter", "Segway", "Ninebot", "ES4", "L
ime-S ES4", 17.2, 30.4, 20.3);
INSERT INTO vehicle_type VALUES (14, "e-bike", "Bike", "Samson", "SLX", "NJS Sam
son", 17.2, 30.4, 20.3);

INSERT INTO allowed_plan VALUES (12, "standard plan");
INSERT INTO allowed_plan VALUES (12, "plan premium plus");
INSERT INTO allowed_plan VALUES (14, "plan premium plus");

INSERT INTO default_plan VALUES (12, "standard plan");
INSERT INTO default_plan VALUES (14, "plan premium plus");

INSERT INTO vehicle VALUES (1, 12, True, False, False, -45.135, -42.135, 20.4, 1
2.5);
INSERT INTO vehicle VALUES (2, 14, True, True, False, -80.135, 100.1354, 73.2, 3
0.4);

INSERT INTO trip VALUES(100, 1, 1, "standard plan", "2022-01-05 01:02:03", "2022
-01-06 01:02:04", -45.135, -42.135, -45.134, -42.146);
INSERT INTO trip VALUES(101, 1, 1, "plan premium plus", "2022-01-07 10:05:33", "
2022-01-08 11:20:04", -45.135, -43.175, -40.134, -42.946);

INSERT INTO trip VALUES(102, 2, 1, "plan premium plus", "2021-07-05 01:02:03", "
2021-07-06 08:04:03", -80.135, -95.135, -80.134, -95.146);
INSERT INTO trip VALUES(103, 2, 2, "plan premium plus", "2022-08-06 07:04:03", "
2022-09-05 01:02:04", -81.135, -95.135, -82.134, -96.146);

-- TODO: add select statements (to print tables)
\! echo 'customer:';
SELECT * FROM customer;
\! echo 'price_plan:';

```

```

SELECT * FROM price_plan;
\! echo 'vehicle_type: ';
SELECT * FROM vehicle_type;
\! echo 'allowed_plan: ';
SELECT * FROM allowed_plan;
\! echo 'default_plan: ';
SELECT * FROM default_plan;
\! echo 'vehicle: ';
SELECT * FROM vehicle;
\! echo 'trip: ';
SELECT * FROM trip;

-- "Breaking" foreign key constraints:

-- 3 Foreign Key Breaks:
-- INSERT INTO default_plan VALUES(14, "standard plan");
-- Note: vehicle_type 14 does not have standard plan as an allowed plan

-- INSERT INTO vehicle VALUES (3, 13, True, False, False, -45.135, -42.135, 20.4
, 12.5);
-- Note: vehicle_type 13 does not exist

-- INSERT INTO trip VALUES(109, 3, 1, "standard plan", "2022-01-05 01:02:03", "2
022-01-06 01:02:04", -45.135, -42.135, -45.134, -42.146);
-- Note: customer 3 does not yet exist

-- 2 Domain Constraints:
-- INSERT INTO vehicle_type VALUES (15, "e-car", "Bike", "Samson", "SLX", "NJS S
amson", 17.2, 30.4, 20.3);
-- Note: vehicle_types can only have form factor "e-scooter" or "e-bike" (not "e
-car")

-- INSERT INTO vehicle VALUES (1, 12, True, False, False, -45.135, -42.135, 101,
12.5);
-- Note: vehicle can only have a fuel percent between 0 and 100 inclusive (not 1
01)

```

Samuel Sovi's Results

customer:

c_id	first_name	last_name	email
1	Samuel	Sovi	samsovi123@gmail.com
2	Daniel	Rodricks	danrodricks123@gmail.com

2 rows in set (0.000 sec)

price_plan:

plan_name	price_per_min	unlock_price	rate_start_min
plan premium plus	5	20	1
standard plan	2	10	3

2 rows in set (0.000 sec)

vehicle_type:

vt_id	form_factor	make	model	trim_level	alt_name	max_spd	max_range	weight
12	e-scooter	Segway	Ninebot	ES4	Lime-S ES4	17.2	30.4	20
14	e-bike	Bike	Samson	SLX	NJS Samson	17.2	30.4	20

2 rows in set (0.000 sec)

allowed_plan:

vt_id	price_plan
12	plan premium plus
12	standard plan
14	plan premium plus

3 rows in set (0.000 sec)

default_plan:

vt_id	price_plan
12	standard plan
14	plan premium plus

2 rows in set (0.000 sec)

vehicle:

v_id	vt_id	in_circ	is_reserved	is_disabled	lat	lon	curr_fuel_pct	cur_range
1	12	1	0	0	-45.135000	-42.135000	20.40	12.5
2	14	1	1	0	-80.135000	100.135400	73.20	30.4

2 rows in set (0.000 sec)

trip:

t_id	c_id	v_id	price_plan	start_dtime	end_dtime	start_lat	start_lon	end_lat	end_lon
100	1	1	standard plan	2022-01-05 01:02:03	2022-01-06 01:02:04	-45.135000	-42.135000	-45.134000	-42.146000
101	1	1	plan premium plus	2022-01-07 10:05:33	2022-01-08 11:20:04	-45.135000	-43.175000	-40.134000	-42.946000
102	2	1	plan premium plus	2021-07-05 01:02:03	2021-07-06 08:04:03	-80.135000	-95.135000	-80.134000	-95.146000
103	2	2	plan premium plus	2022-08-06 07:04:03	2022-09-05 01:02:04	-81.135000	-95.135000	-82.134000	-96.146000

4 rows in set (0.000 sec)

Customer	
PK	c_id
	first_name
	last_name
	email

Trip	
PK	t_id
FK	c_id
FK	v_id
FK	price_plan
	start_dtime
	end_dtime
	start_lat
	start_lon
	end_lat
	end_lon

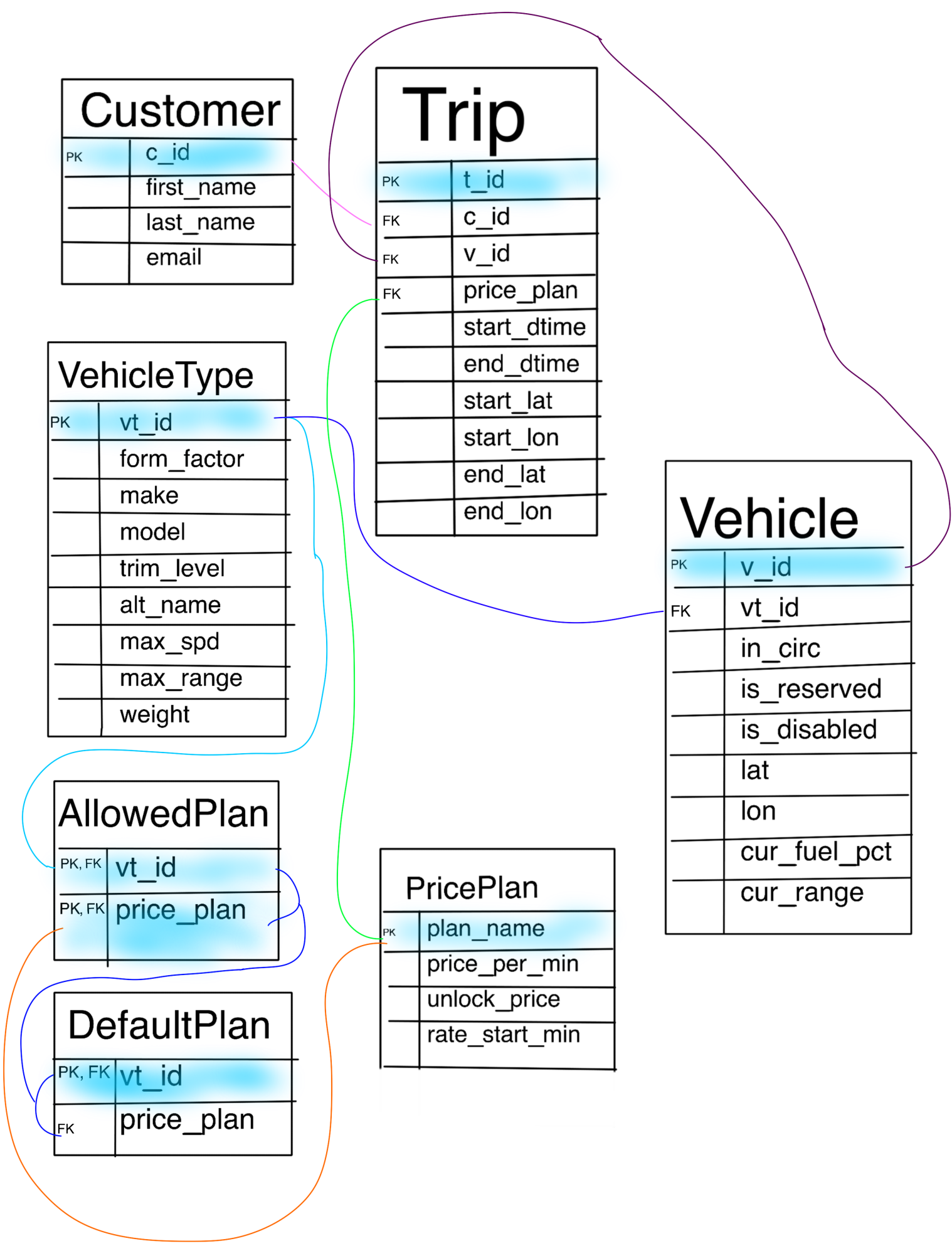
VehicleType	
PK	vt_id
	form_factor
	make
	model
	trim_level
	alt_name
	max_spd
	max_range
	weight

Vehicle	
PK	v_id
FK	vt_id
	in_circ
	is_reserved
	is_disabled
	lat
	lon
	cur_fuel_pct
	cur_range

AllowedPlan	
PK, FK	vt_id
PK, FK	price_plan

PricePlan	
PK	plan_name
	price_per_min
	unlock_price
	rate_start_min

DefaultPlan	
PK, FK	vt_id
FK	price_plan



Foreign Key “Breaks”

```
INSERT INTO default_plan VALUES(14, "standard plan");  
-- Note: vehicle_type 14 does not have standard plan as an allowed plan
```

```
ERROR 1452 (23000) at line 66 in file: 'hw1-data.sql': Cannot add or update a child row: a foreign key constraint fails ('ssovi_DB`.`default_plan`, CONSTRAINT `default_plan_ibfk_1` FOREIGN KEY (`vt_id`, `price_plan`) REFERENCES `allowed_plan` (`vt_id`, `price_plan`))
```

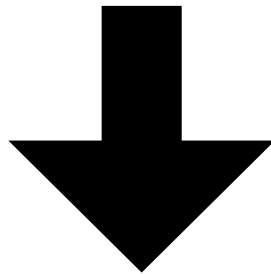
```
INSERT INTO vehicle VALUES (3, 13, True, False, False, -45.135, -42.135, 20.4, 12.5);  
-- Note: vehicle_type 13 does not exist
```

```
ERROR 1452 (23000) at line 70 in file: 'hw1-data.sql': Cannot add or update a child row: a foreign key constraint fails ('ssovi_DB`.`vehicle`, CONSTRAINT `vehicle_ibfk_1` FOREIGN KEY (`vt_id`) REFERENCES `vehicle_type` (`vt_id`))
```

```
INSERT INTO trip VALUES(109, 3, 1, "standard plan", "2022-01-05 01:02:03", "2022-01-06 01:02:04", -45.135, -42.135, -45.134, -42.146);  
-- Note: customer 3 does not yet exist
```

```
ERROR 1452 (23000) at line 73 in file: 'hw1-data.sql': Cannot add or update a child row: a foreign key constraint fails ('ssovi_DB`.`trip`, CONSTRAINT `trip_ibfk_1` FOREIGN KEY (`c_id`) REFERENCES `customer` (`c_id`))
```

Domain “Breaks” on next page



Domain Constraint “Breaks”

```
-- Domain constraint  
INSERT INTO vehicle_type VALUES (15, "e-car", "Bike", "Samson", "SLX", "NJS Samson", 17.2, 30.4, 20.3);  
-- Note: vehicle_types can only have form factor "e-scooter" or "e-bike" (not "e-car")
```

```
ERROR 4025 (23000) at line 78 in file: 'hw1-data.sql': CONSTRAINT `valid_form_factor` failed for `ssovi_DB`.`vehicle_type`  
MySQL [ssovi_DB]:
```

```
INSERT INTO vehicle VALUES (1, 12, True, False, False, -45.135, -42.135, 101, 12.5);  
-- Note: vehicle can only have a fuel percent between 0 and 100 inclusive (not 101)
```

```
ERROR 4025 (23000) at line 81 in file: 'hw1-data.sql': CONSTRAINT `valid_curr_fuel_pct` failed for `ssovi_DB`.`vehicle`  
MySQL [ssovi_DB]:
```