Creating a custom data type

Apache Cassandra allows for the creation of custom **user-defined types** (**UDTs**). UDTs allow for further denormalization of data within a row. A good example of this is a mailing address for customers. Assume a simple table:

```
CREATE TABLE customer (
 last_name TEXT,
 first_name TEXT,
 company TEXT,
 PRIMARY KEY (last_name,first_name));
```

Now, our customers have mailing addresses. Corporate customers usually have addresses for multiple things, including billing, shipping, headquarters, distribution centers, store locations, and data centers. So how do we track multiple addresses for a single customer? One way to accomplish this would be to create a collection of a UDT:

```
CREATE TYPE customer_address (
 type TEXT,
 street TEXT,
 city TEXT,
 state TEXT,
 postal_code TEXT,
 country TEXT);
```

Now, let's add the `customer_address` UDT to the table as a list. This way, a customer can have multiple addresses:

```
ALTER TABLE customer ADD addresses LIST<FROZEN <customer_address>>;
```

Note

The `FROZEN` types are those that are immutable. They are written once and cannot be changed, short of rewriting all underlying properties.

With that in place, let's add a few rows to the table:

```
 INSERT INTO customer (last_name,first_name,company,addresses) VALUES ('Washburne','Hoban','Serenity',[{type:'SHIPPING',street:'9843
INSERT INTO customer (last_name,first_name,company,addresses) VALUES ('Washburne','Zoey','Serenity',[{type:'SHIPPING',street:'9843 3
INSERT INTO customer (last_name,first_name,company,addresses) VALUES ('Tam','Simon','Persephone General Hospital',[{type:'BILL TO',s
```

Querying it for Zoey Washburne shows that her company has two addresses:

```
 SELECT last_name,first_name,company,addresses FROM customer
WHERE last_name='Washburne' AND first_name='Zoey';

last_name | first_name |  company | addresses
-----------+------------+----------------------------------------------------------------------------------------------------------
Washburne |       Zoey | Serenity |[{type: 'SHIPPING', street: '9843 32nd Place', city: 'Charlotte', state: 'NC', postal_code: '0560

(1 rows)
```

Altering a custom type

UDTs can have columns added to them. For instance, some addresses have two lines, so we can add an `address2` column:

```
ALTER TYPE customer_address ADD address2 TEXT;
```

UDT columns can also be renamed with the `ALTER` command:

```
ALTER TYPE customer_address RENAME address2 TO street2;
```

Note

Columns within a UDT cannot be removed or dropped, only added or renamed.

Dropping a custom type

UDTs can be dropped very easily, just by issuing the `DROP` command. If we create a simple UDT:

```
CREATE TYPE test_type (value TEXT);
```

It can be dropped like this:

```
DROP TYPE test_type;
```