```
 1    -- Demonstrates deleting rows from a single table
 2    -- Uses mfa.db
 3
 4    -- Deletes item with particular title
 5    DELETE FROM "collections" WHERE "title" = 'Spring outing';
 6
 7    -- Deletes item where value is NULL
 8    DELETE FROM "collections" WHERE "acquired" IS NULL;
 9
10    -- Deletes items acquired before the museum moved to a new location in 1909
11    DELETE FROM "collections" WHERE "acquired" < '1909-01-01';
```

```sql
-- Demonstrates deleting rows with constraints
-- Uses mfa.db

-- Raises a foreign key constraint error
DELETE FROM "artists" WHERE "name" = 'Unidentified artist';

-- Deletes the artist's affiliation with their work, using hard-coded id
DELETE FROM "created" WHERE "artist_id" = 3;

-- Deletes the artist's affiliation with their work, using subquery
DELETE FROM "created" WHERE "artist_id" = (
    SELECT "id" FROM "artists" WHERE "name" = 'Unidentified artist'
);

-- Deletes the artist themselves
DELETE FROM "artists" WHERE "name" = 'Unidentified artist';
```

```sql
1  -- Demonstrates deleting rows with ON DELETE actions
2  -- Uses mfa.db
3
4  -- Deletes an artist when foreign key ON DELETE action is set to CASCADE
5  DELETE FROM "artists" WHERE "name" = 'Unidentified artist';
```

```sql
-- Demonstrates schema of mfa.db for delete files
-- Creates mfa.db

-- Deletes prior tables if they exist
DROP TABLE IF EXISTS "collections";
DROP TABLE IF EXISTS "artists";
DROP TABLE IF EXISTS "created";

-- Creates new table and add data
CREATE TABLE "collections" IF NOT EXISTS (
    "id" INTEGER,
    "title" TEXT NOT NULL,
    "accession_number" TEXT NOT NULL UNIQUE,
    "acquired" NUMERIC,
    PRIMARY KEY("id")
);

INSERT INTO "collections" ("title", "accession_number", "acquired")
VALUES
('Profusion of flowers', '56.257', '1956-04-12'),
('Farmers working at dawn', '11.6152', '1911-08-03'),
('Spring outing', '14.76', '1914-01-08'),
('Imaginative landscape', '56.496', NULL),
('Peonies and butterfly', '06.1899', '1906-01-01'),
('Tile Lunette', '06.2437', '1906-11-08'),
('Statuette of a shrew', '01.105', '1901-02-11');

-- Tile Lunette: https://collections.mfa.org/objects/8559/tile-lunette?ctx=0510197c-0c9b-4a77-93ea-f5246427388a&idx=3
-- Statuette of a shrew: https://collections.mfa.org/objects/132432/statuette-of-a-shrew?ctx=3247cc4b-30db-4047-b9c7-813a6d39e11c&idx=5
```

```sql
-- Adds additional tables to provoke foreign key constraints
-- Creates mfa.db

-- Deletes prior tables if they exist
DROP TABLE IF EXISTS "collections";
DROP TABLE IF EXISTS "artists";
DROP TABLE IF EXISTS "created";

-- Creates new tables and add data
CREATE TABLE "collections" IF NOT EXISTS (
    "id" INTEGER,
    "title" TEXT NOT NULL,
    "accession_number" TEXT NOT NULL UNIQUE,
    "acquired" NUMERIC,
    PRIMARY KEY("id")
);

INSERT INTO "collections" ("title", "accession_number", "acquired")
VALUES
('Farmers working at dawn', '11.6152', '1911-08-03'),
('Imaginative landscape', '56.496', NULL),
('Profusion of flowers', '56.257', '1956-04-12'),
('Spring outing', '14.76', '1914-01-08');

CREATE TABLE "artists" (
    "id" INTEGER,
    "name" TEXT NOT NULL,
    PRIMARY KEY("id")
);

INSERT INTO "artists" ("name")
VALUES
('Li Yin'),
('Qian Weicheng'),
('Unidentified artist'),
('Zhou Chen');

CREATE TABLE "created" (
    "artist_id" INTEGER,
    "collection_id" INTEGER,
    PRIMARY KEY("artist_id", "collection_id"),
    FOREIGN KEY("artist_id") REFERENCES "artists"("id"),
```

```
43        FOREIGN KEY("collection_id") REFERENCES "collections"("id")
44  );
45
46  INSERT INTO "created" ("artist_id", "collection_id")
47  VALUES
48  ((SELECT "id" FROM "artists" WHERE "name" = 'Li Yin'), (SELECT "id" FROM "collections" WHERE "title" = 'Imaginative landscape')),
49  ((SELECT "id" FROM "artists" WHERE "name" = 'Qian Weicheng'), (SELECT "id" FROM "collections" WHERE "title" = 'Profusion of flowers')),
50  ((SELECT "id" FROM "artists" WHERE "name" = 'Unidentified artist'), (SELECT "id" FROM "collections" WHERE "title" = 'Farmers working at
dawn')),
51  ((SELECT "id" FROM "artists" WHERE "name" = 'Zhou Chen'), (SELECT "id" FROM "collections" WHERE "title" = 'Spring outing'));
```

```sql
-- Adds ON DELETE action
-- Creates mfa.db

-- Deletes prior tables if they exist
DROP TABLE IF EXISTS "collections";
DROP TABLE IF EXISTS "artists";
DROP TABLE IF EXISTS "created";

-- Creates new tables and add data
CREATE TABLE "collections" IF NOT EXISTS (
    "id" INTEGER,
    "title" TEXT NOT NULL,
    "accession_number" TEXT NOT NULL UNIQUE,
    "acquired" NUMERIC,
    PRIMARY KEY("id")
);

INSERT INTO "collections" ("title", "accession_number", "acquired")
VALUES
('Farmers working at dawn', '11.6152', '1911-08-03'),
('Imaginative landscape', '56.496', NULL),
('Profusion of flowers', '56.257', '1956-04-12'),
('Spring outing', '14.76', '1914-01-08');

CREATE TABLE "artists" IF NOT EXISTS (
    "id" INTEGER,
    "name" TEXT NOT NULL,
    PRIMARY KEY("id")
);

INSERT INTO "artists" ("name")
VALUES
('Li Yin'),
('Qian Weicheng'),
('Unidentified artist'),
('Zhou Chen');

CREATE TABLE "created" IF NOT EXISTS (
    "artist_id" INTEGER,
    "collection_id" INTEGER,
    PRIMARY KEY("artist_id", "collection_id"),
    FOREIGN KEY("artist_id") REFERENCES "artists"("id") ON DELETE CASCADE,
```

```sql
43          FOREIGN KEY("collection_id") REFERENCES "collections"("id") ON DELETE CASCADE
44     );
45
46     INSERT INTO "created" ("artist_id", "collection_id")
47     VALUES
48     ((SELECT "id" FROM "artists" WHERE "name" = 'Li Yin'), (SELECT "id" FROM "collections" WHERE "title" = 'Imaginative landscape')),
49     ((SELECT "id" FROM "artists" WHERE "name" = 'Qian Weicheng'), (SELECT "id" FROM "collections" WHERE "title" = 'Profusion of flowers')),
50     ((SELECT "id" FROM "artists" WHERE "name" = 'Unidentified artist'), (SELECT "id" FROM "collections" WHERE "title" = 'Farmers working at
dawn')),
51     ((SELECT "id" FROM "artists" WHERE "name" = 'Zhou Chen'), (SELECT "id" FROM "collections" WHERE "title" = 'Spring outing'));
```

```sql
1   -- Demonstrates soft deletes
2   -- Uses mfa.db
3
4   -- Adds a "deleted" column to "collections" table
5   ALTER TABLE "collections" ADD COLUMN "deleted" INTEGER DEFAULT 0;
6
7   -- Views updated schema of collections table
8   .schema "collections"
9
10  -- Views data
11  SELECT * FROM "collections";
12
13  -- Instead of deleting an item, updates its deleted column to be 1
14  UPDATE "collections" SET "deleted" = 1 WHERE "title" = 'Farmers working at dawn';
15
16  -- Selects all items from collections that are not deleted
17  SELECT * FROM "collections" WHERE "deleted" != 1;
```

```
 1   -- Demonstrates importing a CSV into an existing table
 2   -- Creates mfa.db
 3
 4   -- Deletes prior tables if they exist
 5   DROP TABLE IF EXISTS "collections";
 6   DROP TABLE IF EXISTS "artists";
 7   DROP TABLE IF EXISTS "created";
 8
 9   -- Imports into an existing table by specifying mode and skipping header columns
10   .import --csv --skip 1 mfa.csv collections
```

```sql
1    -- Demonstrates importing a CSV into an existing table and adding primary keys
2    -- Creates mfa.db
3
4    -- Deletes prior tables if they exist
5    DROP TABLE IF EXISTS "collections";
6    DROP TABLE IF EXISTS "artists";
7    DROP TABLE IF EXISTS "created";
8
9    -- Imports into a temporary using a CSV without primary keys
10   .import --csv mfa.csv temp
11
12   -- Creates a final table to store imported data
13   CREATE TABLE "collections" IF NOT EXISTS (
14       "id" INTEGER,
15       "title" TEXT NOT NULL,
16       "accession_number" TEXT NOT NULL UNIQUE,
17       "acquired" NUMERIC,
18       PRIMARY KEY("id")
19   );
20
21   -- Inserts into final table by selecting from temporary table
22   INSERT INTO "collections" ("title", "accession_number", "acquired")
23   SELECT "title", "accession_number", "acquired" FROM "temp";
24
25   -- Deletes temporary table
26   DROP TABLE "temp";
```

```sql
1   -- Demonstrates adding individual rows to a table
2   -- Uses mfa.db
3
4   -- Adds a new item to the collections
5   INSERT INTO "collections" ("id", "title", "accession_number", "acquired")
6   VALUES (1, 'Profusion of flowers', '56.257', '1956-04-12');
7
8   -- Adds a new item to the collections
9   INSERT INTO "collections" ("id", "title", "accession_number", "acquired")
10  VALUES (2, 'Farmers working at dawn', '11.6152', '1911-08-03');
11
12  -- Adds a new item to the collections, demonstrating primary key auto-increments
13  INSERT INTO "collections" ("title", "accession_number", "acquired")
14  VALUES ('Spring outing', '14.76', '1914-01-08');
15
16  -- Shows violation of UNIQUE
17  INSERT INTO "collections" ("title", "accession_number", "acquired")
18  VALUES ('Spring outing', '14.76', '1914-01-08');
19
20  -- Shows violation of NOT NULL
21  INSERT INTO "collections" ("title", "accession_number", "acquired")
22  VALUES (NULL, '56.496', '1914-01-08');
23
24  -- Profusion of flowers: https://collections.mfa.org/objects/254/profusion-of-flowers?ctx=59408041-a021-4b91-bceb-580fd6fe7e17&idx=5
25  -- Farmers working at dawn: https://collections.mfa.org/objects/256/farmers-working-at-dawn?ctx=59408041-a021-4b91-bceb-580fd6fe7e17&idx=7
26  -- Spring outing: https://collections.mfa.org/objects/353/spring-outing?ctx=87931f50-caf4-4309-8175-96c5196e52bb&idx=23
```

```sql
1   -- Demonstrates adding multiple rows to a table
2   -- Uses mfa.db
3
4   -- Adds a set of new items to the collection
5   INSERT INTO "collections" ("title", "accession_number", "acquired")
6   VALUES
7   ('Imaginative landscape', '56.496', NULL),
8   ('Peonies and butterfly', '06.1899', '1906-01-01');
9
10  -- Imaginative landscape: https://collections.mfa.org/objects/318/imaginative-landscape?ctx=792f664e-1945-4e0e-9c34-1de7d932bf36&idx=14
11  -- Peonies and butterfly: https://collections.mfa.org/objects/8532/peonies-and-butterfly?ctx=26ebba4c-6b5c-4234-8e01-60eba690cd70&idx=65
```

```sql
-- Demonstrates schema of mfa.db for insert files
-- Creates mfa.db

-- Deletes prior tables if they exist
DROP TABLE IF EXISTS "collections";
DROP TABLE IF EXISTS "artists";
DROP TABLE IF EXISTS "created";

-- Creates collections table
CREATE TABLE "collections" IF NOT EXISTS (
    "id" INTEGER,
    "title" TEXT NOT NULL,
    "accession_number" TEXT NOT NULL UNIQUE,
    "acquired" NUMERIC,
    PRIMARY KEY("id")
);
```

```sql
-- Demonstrates schema of mfa.db for trigger files
-- Creates mfa.db

-- Deletes prior tables if they exist
DROP TABLE IF EXISTS "collections";
DROP TABLE IF EXISTS "artists";
DROP TABLE IF EXISTS "created";

CREATE TABLE "collections" IF NOT EXISTS (
    "id" INTEGER,
    "title" TEXT NOT NULL,
    "accession_number" TEXT NOT NULL UNIQUE,
    "acquired" NUMERIC,
    PRIMARY KEY("id")
);

INSERT INTO "collections" ("title", "accession_number", "acquired")
VALUES
('Farmers working at dawn', '11.6152', '1911-08-03'),
('Imaginative landscape', '56.496', NULL),
('Profusion of flowers', '56.257', '1956-04-12'),
('Spring outing', '14.76', '1914-01-08');

CREATE TABLE "artists" IF NOT EXISTS (
    "id" INTEGER,
    "name" TEXT NOT NULL,
    PRIMARY KEY("id")
);

INSERT INTO "artists" ("name")
VALUES
('Li Yin'),
('Qian Weicheng'),
('Unidentified artist'),
('Zhou Chen');

CREATE TABLE "created" IF NOT EXISTS (
    "artist_id" INTEGER,
    "collection_id" INTEGER,
    PRIMARY KEY("artist_id", "collection_id"),
    FOREIGN KEY("artist_id") REFERENCES "artists"("id") ON DELETE CASCADE,
    FOREIGN KEY("collection_id") REFERENCES "collections"("id") ON DELETE CASCADE
```

```
43    );
44
45    INSERT INTO "created" ("artist_id", "collection_id")
46    VALUES
47    ((SELECT "id" FROM "artists" WHERE "name" = 'Li Yin'), (SELECT "id" FROM "collections" WHERE "title" = 'Imaginative landscape')),
48    ((SELECT "id" FROM "artists" WHERE "name" = 'Qian Weicheng'), (SELECT "id" FROM "collections" WHERE "title" = 'Profusion of flowers')),
49    ((SELECT "id" FROM "artists" WHERE "name" = 'Unidentified artist'), (SELECT "id" FROM "collections" WHERE "title" = 'Farmers working at
dawn')),
50    ((SELECT "id" FROM "artists" WHERE "name" = 'Zhou Chen'), (SELECT "id" FROM "collections" WHERE "title" = 'Spring outing'));
```

```sql
 1   -- Demonstrates triggers on delete and insert
 2   -- Uses mfa.db
 3
 4   -- Creates a table to track buying and selling of items from collections
 5   CREATE TABLE "transactions" (
 6       "id" INTEGER,
 7       "title" TEXT,
 8       "action" TEXT,
 9       PRIMARY KEY("id")
10   );
11
12   -- Creates a trigger to log selling items from collections
13   CREATE TRIGGER "sell"
14   BEFORE DELETE ON "collections"
15   BEGIN
16       INSERT INTO "transactions" ("title", "action")
17       VALUES (OLD."title", 'sold');
18   END;
19
20   -- Lists existing triggers
21   .schema
22
23   -- Deletes from collections
24   DELETE FROM "collections" WHERE "title" = 'Profusion of flowers';
25
26   -- Creates a trigger to log buying items
27   CREATE TRIGGER "buy"
28   AFTER INSERT ON "collections"
29   BEGIN
30       INSERT INTO "transactions" ("title", "action")
31       VALUES (NEW."title", 'bought');
32   END;
33
34   -- Adds item to collections
35   INSERT INTO "collections" ("title", "accession_number", "acquired")
36   VALUES ('Profusion of flowers', '56.257', '1956-04-12');
```

```sql
1   -- Demonstrates schema of mfa.db for update files
2   -- Creates mfa.db
3
4   -- Deletes prior tables if they exist
5   DROP TABLE IF EXISTS "collections";
6   DROP TABLE IF EXISTS "artists";
7   DROP TABLE IF EXISTS "created";
8
9   CREATE TABLE "collections" IF NOT EXISTS (
10      "id" INTEGER,
11      "title" TEXT NOT NULL,
12      "accession_number" TEXT NOT NULL UNIQUE,
13      "acquired" NUMERIC,
14      PRIMARY KEY("id")
15  );
16
17  INSERT INTO "collections" ("title", "accession_number", "acquired")
18  VALUES
19  ('Farmers working at dawn', '11.6152', '1911-08-03'),
20  ('Imaginative landscape', '56.496', NULL),
21  ('Profusion of flowers', '56.257', '1956-04-12'),
22  ('Spring outing', '14.76', '1914-01-08');
23
24  CREATE TABLE "artists" IF NOT EXISTS (
25      "id" INTEGER,
26      "name" TEXT NOT NULL,
27      PRIMARY KEY("id")
28  );
29
30  INSERT INTO "artists" ("name")
31  VALUES
32  ('Li Yin'),
33  ('Qian Weicheng'),
34  ('Unidentified artist'),
35  ('Zhou Chen');
36
37  CREATE TABLE "created" IF NOT EXISTS (
38      "artist_id" INTEGER,
39      "collection_id" INTEGER,
40      PRIMARY KEY("artist_id", "collection_id"),
41      FOREIGN KEY("artist_id") REFERENCES "artists"("id") ON DELETE CASCADE,
42      FOREIGN KEY("collection_id") REFERENCES "collections"("id") ON DELETE CASCADE
```

```sql
43    );
44
45    INSERT INTO "created" ("artist_id", "collection_id")
46    VALUES
47    ((SELECT "id" FROM "artists" WHERE "name" = 'Li Yin'), (SELECT "id" FROM "collections" WHERE "title" = 'Imaginative landscape')),
48    ((SELECT "id" FROM "artists" WHERE "name" = 'Qian Weicheng'), (SELECT "id" FROM "collections" WHERE "title" = 'Profusion of flowers')),
49    ((SELECT "id" FROM "artists" WHERE "name" = 'Unidentified artist'), (SELECT "id" FROM "collections" WHERE "title" = 'Farmers working at
dawn')),
50    ((SELECT "id" FROM "artists" WHERE "name" = 'Zhou Chen'), (SELECT "id" FROM "collections" WHERE "title" = 'Spring outing'));
```

```sql
1    -- Demonstrates updating authorship
2    -- Uses mfa.db
3
4    -- Updates authorship (incorrectly)
5    UPDATE "created" SET "artist_id" = (
6        SELECT "id" FROM "artists"
7        WHERE "name" = 'Li Yin'
8    );
9
10   -- Updates authorship (correctly) for a piece with a previously unknown authorship
11   UPDATE "created" SET "artist_id" = (
12       SELECT "id" FROM "artists"
13       WHERE "name" = 'Li Yin'
14   )
15   WHERE "collection_id" = (
16       SELECT "id" FROM "collections"
17       WHERE "title" = 'Farmers working at dawn'
18   );
```

```sql
-- Demonstrates cleaning data from a CSV of votes for favorite artwork
-- Creates votes.db

-- Imports votes.csv
.import votes.csv votes

-- Counts votes
SELECT "title", COUNT("title") FROM "votes" GROUP BY "title";

-- Removes trailing whitespace
UPDATE "votes" SET "title" = trim("title");

-- Forces to uppercase
UPDATE "votes" SET "title" = upper("title");

-- Manually updates the titles of "Farmers working at dawn"
UPDATE "votes" SET "title" = 'FARMERS WORKING AT DAWN'
WHERE "title" = 'FARMERS WORKING';

UPDATE "votes" SET "title" = 'FARMERS WORKING AT DAWN'
WHERE "title" = 'FAMERS WORKING AT DAWN';

-- Fixes misspellings of "Farmers working at dawn"
UPDATE "votes" SET "title" = 'FARMERS WORKING AT DAWN'
WHERE "title" LIKE 'Fa%';

-- Fixes misspellings of "Imaginative landscape"
UPDATE "votes" SET "title" = 'IMAGINATIVE LANDSCAPE'
WHERE "title" LIKE 'Imag%';

-- Fixes misspellings of "Profusion of flowers"
UPDATE "votes" SET "title" = 'PROFUSION OF FLOWERS'
WHERE "title" LIKE 'Profusion %';
```