

**select \* from hw5\_original;**

**DROP TABLE IF EXISTS artists;**  
**CREATE TABLE artists(**  
    **artistId INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,**  
    **artistName VARCHAR(20) unique not NULL**  
**);**

**DROP TABLE IF EXISTS albums;**  
**CREATE TABLE albums (**  
    **albumId INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,**  
    **albumTitle VARCHAR(255),**  
    **artistId INT,**  
    **FOREIGN KEY (artistId) REFERENCES artists(artistId)**  
**);**

**DROP TABLE IF EXISTS media\_types;**  
**CREATE TABLE media\_types (**  
    **mediaTypeId INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,**  
    **mediaName VARCHAR(20) not NULL**  
**);**

**DROP TABLE IF EXISTS genres;**  
**CREATE TABLE genres (**  
    **genreId INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,**  
    **genreName VARCHAR(20) not NULL**  
**);**

**DROP TABLE IF EXISTS Customers;**  
**CREATE TABLE Customers (**  
    **customer\_id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,**  
    **firstName VARCHAR(255) not NULL,**  
    **lastName VARCHAR(255) not NULL,**  
    **address VARCHAR(255),**  
    **city VARCHAR(255),**  
    **state VARCHAR(255),**  
    **country VARCHAR(255),**  
    **postalCode VARCHAR(255),**  
    **phoneNumber VARCHAR(255),**  
    **faxNumber VARCHAR(255),**  
    **email VARCHAR(255),**  
    **UNIQUE(lastName) on CONFLICT ABORT**  
**);**

**DROP TABLE IF EXISTS Invoices;**  
**CREATE TABLE Invoices (**  
    **InvoiceId INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,**  
    **Date TEXT not NULL,**  
    **billingAddress VARCHAR(255),**  
    **billingCity VARCHAR(255),**  
    **billingState VARCHAR(255),**  
    **billingCountry VARCHAR(255),**  
    **billingPostalCode VARCHAR(255),**  
    **customer\_id INT,**  
    **FOREIGN KEY (customer\_id) REFERENCES Customers(customer\_id),**

```

        UNIQUE(InvoiceId) on conflict abort
    );

DROP TABLE IF EXISTS tracks;
CREATE TABLE tracks (
    trackId INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    trackName VARCHAR(255) not NULL,
    composer VARCHAR(255),
    trackSizeBytes INT,
    trackLength INT not NULL,
    trackprice DECIMAL(10,2) not NULL,
    genreId INT,
    mediaTypeId INT,
    albumId INT,
    FOREIGN KEY (mediaTypeId) REFERENCES media_types(mediaTypeId),
    FOREIGN KEY (genreId) REFERENCES genres(genreId),
    FOREIGN KEY (albumId) REFERENCES albums(albumId)
);

DROP TABLE IF EXISTS Invoice_items;
CREATE TABLE Invoice_items (
    InvoiceItemID INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    trackId INT,
    quantity INT not NULL,
    unitPrice DECIMAL(10,2) not NULL,
    InvoiceId INT,
    FOREIGN KEY (InvoiceId) REFERENCES Invoices(InvoiceId),
    FOREIGN KEY (trackId) REFERENCES tracks(trackId)
);

INSERT INTO media_types(mediaName)
SELECT DISTINCT MediaType
FROM hw5_original;

select * from media_types;

INSERT INTO genres(genreName)
SELECT DISTINCT Genre
FROM hw5_original;

select * from genres;

INSERT INTO artists(artistName)
SELECT DISTINCT artistName
FROM hw5_original;

select * from artists;

INSERT INTO albums(albumTitle, artistId)
SELECT DISTINCT albumTitle, artistId
FROM hw5_original h join artists a on h.artistName = a.artistName;

select * from albums;

```

```

INSERT INTO Customers(firstName, lastName, address, city, state, country, postalCode,
phoneNumber, faxNumber, email)
SELECT DISTINCT
    CustomerFirstName,
    CustomerLastName,
    CustomerAddress,
    CustomerCity,
    CustomerState,
    CustomerCountry,
    CustomerPostalCode,
    CustomerPhone,
    CustomerFax,
    CustomerEmail
FROM hw5_original
where CustomerFirstName is not null;

```

```

select * from Customers;

```

```

INSERT INTO Invoices(InvoiceId, Date, billingAddress, billingCity, billingState, billingCountry,
billingPostalCode, customer_id)
SELECT DISTINCT
    InvoiceId,
    InvoiceDate,
    InvoiceBillingAddress,
    InvoiceBillingCity,
    InvoiceBillingState,
    InvoiceBillingCountry,
    InvoiceBillingPostalCode,
    customer_id
FROM hw5_original h
join Customers c on h.CustomerFirstName = c.firstName AND h.CustomerLastName = c.lastName
;

```

```

select * from Invoices;

```

```

INSERT INTO tracks(trackName, composer, trackSizeBytes, trackLength, trackprice, genreId,
mediaTypeId, albumId)
select DISTINCT
    TrackName,
    Composer,
    trackSizeBytes,
    trackLength,
    TrackPrice,
    genreId,
    mediaTypeId,
    albumId
from hw5_original h
JOIN genres AS g ON h.Genre = g.genreName
JOIN media_types AS m ON h.MediaType = m.mediaName
JOIN albums AS a ON h.AlbumTitle = a.albumTitle;

```

```

select * from tracks;

```

```

INSERT INTO Invoice_items(InvoiceId, trackId, quantity, unitPrice)
select DISTINCT

```

```
        i.InvoiceID,  
        t.trackId,  
        h.InvoiceItemQuantity,  
        h.InvoiceItemUnitPrice  
FROM      hw5_original AS h  
JOIN      tracks AS t ON t.trackName = h.trackName AND t.trackSizeBytes = h.TrackSizeBytes  
JOIN      Invoices AS i ON h.InvoiceId = i.InvoiceId  
WHERE     h.InvoiceID IS NOT NULL;  
  
select * from Invoice_items;
```