

-- Customer Table Creation

```
create table customers(  
    customer_id SERIAL primary key,  
    first_name VARCHAR(100),  
    last_name VARCHAR(150),  
    address VARCHAR(100),  
    city VARCHAR(100),  
    state VARCHAR(50),  
    zip_code VARCHAR(50),  
    phone_number VARCHAR(15),  
    email_address VARCHAR(150)  
);
```

-- Concessions Table Creation

```
create table concessions(  
    concession_id SERIAL primary key,  
    concession_item_name VARCHAR(100),  
    concession_item_cost NUMERIC(3,2),  
    concession_item_purchase_date DATE default CURRENT_DATE,  
    concession_payment_method VARCHAR(50),  
    concession_payment_credit_card_type VARCHAR(50),  
    concession_payment_credit_card_number VARCHAR(16),  
    concession_payment_credit_card_expiration VARCHAR(15),  
    concession_payment_credit_card_security_code VARCHAR(3),  
    customer_id INTEGER not null,  
    foreign key(customer_id) references customers(customer_id)  
);
```

-- Movies Table Creation

```
create table movies(  
    movie_id SERIAL primary key,  
    movie_name VARCHAR(150),  
    movie_release_date DATE,  
    movie_rating VARCHAR(50),  
    movie_genre VARCHAR(100),  
    movie_runtime VARCHAR(15)  
);
```

-- Tickets Table Creation

```
create table tickets(  
    ticket_purchase_id SERIAL primary key,  
    ticket_purchase_date DATE default CURRENT_DATE,  
    ticket_cost_subtotal NUMERIC(4,2),  
    ticket_cost_total NUMERIC(4,2),  
    ticket_movie_date DATE,  
    ticket_payment_method VARCHAR(50),  
    ticket_payment_credit_card_type VARCHAR(50),  
    ticket_payment_credit_card_number VARCHAR(16),  
    ticket_payment_credit_card_expiration VARCHAR(15),  
    ticket_payment_credit_card_security_code VARCHAR(3),  
    customer_id INTEGER not null,  
    movie_id INTEGER not null,  
    foreign key(customer_id) references customers(customer_id),  
    foreign key(movie_id) references movies(movie_id)  
);
```