CS-GY 6083 B - Principles of Database Systems Report for Project Part II

Section: B

Date: May 11, 2023

Participants:

Sai Krishna Chaitanya Annavazala – sa6948

Sai Indukuri – si2220

Sai Pranaswi Mullangi – sm11006

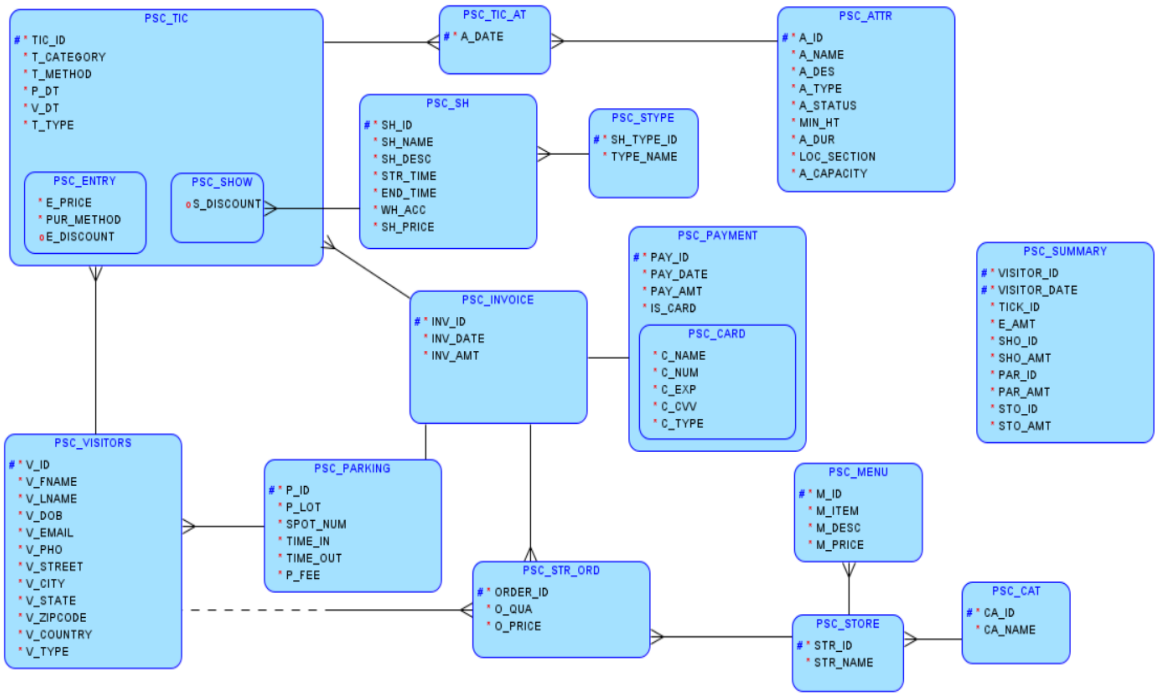
Table of Contents

Executive Summary

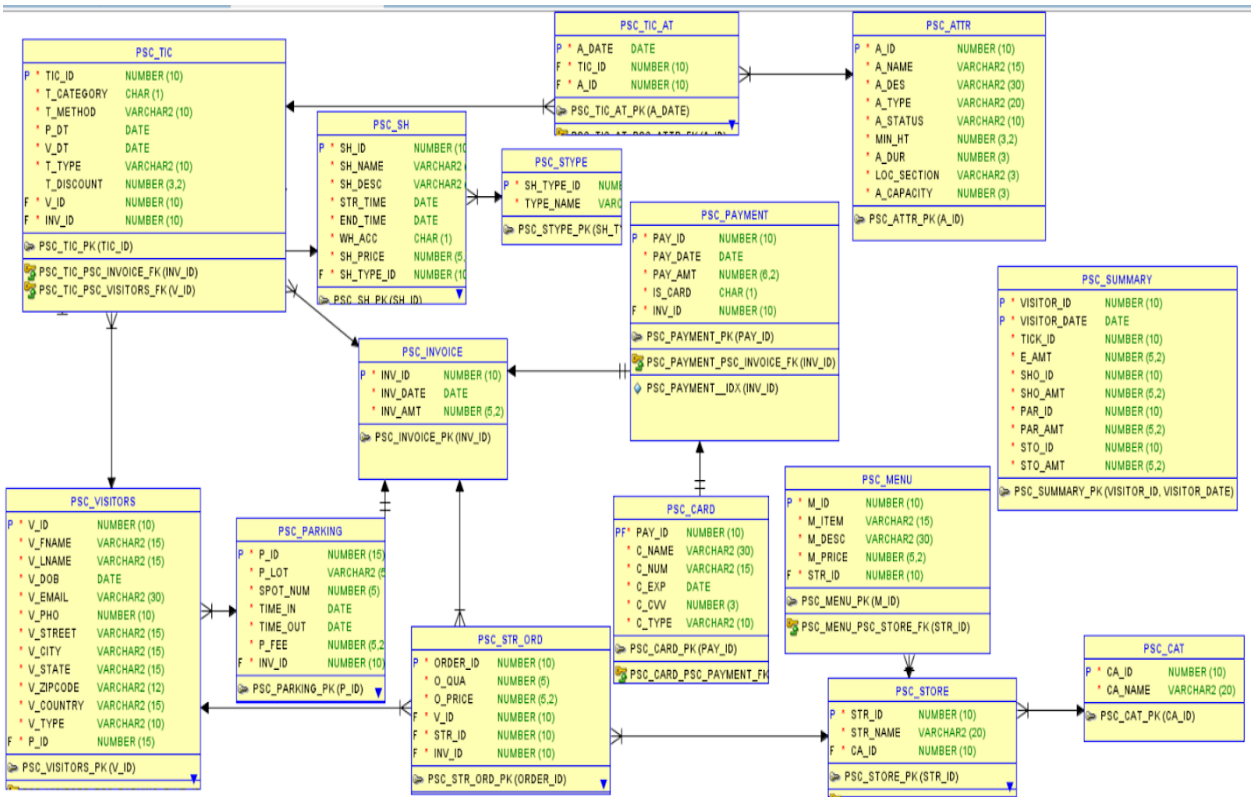
In this project, our aim is to create a web-based user interface for the database schema from the first part of the project. For this, we implemented a web application on the concept of theme park that provides various kinds of entertainment to people of all ages. Theme Park has different attractions, shows, and stores for the visitors. It also has a parking facility for visitors. Our application allows the visitors to book the tickets and pay online. It stores the database of all the bookings, payments, and visitors to each entertainment.

In addition to the services mentioned above the theme park would also allow visitors to sign up and login to their account and manage access authorizations of diverse groups of users.

Logical Model



Relational Model



Assumptions

- Once entering the theme park, all the invoices generated would be paid when the visitor exits the park.
- The entire amount would be paid in a single payment. So, one-one relationship between invoice and payment.
- One most important assumption is that once a visitor enters a theme park, he has 2 choices of watching and not watching a show. So, show tickets are taken separately and show price is calculated seperately.
- One parking lot is given to many visitors because later visitors may enter after the former visitor leaves.
- For a single show type (ex:drama) there are many shows and for each show there are many tickets.
- For a single visitor there are many tickets because he may take a ticket for both entry and show or a single visitor may visit more than one time in a day.
- For a store type there will be many stores and for a single store there will be many orders and many items.
- Single visitor is assumed to be given many orders.
- One invoice has one parking fee, many ticket prices and many order prices. Here everything is stored in a single PSC_INVOICE entity by giving appropriate primary and foreign keys.

Tech Stack

Front end: Django, HTML, CSS

Server side backend: Node.js

Database: MySQL

We used the django framework in the Python programming language to facilitate rapid prototyping of the application with tons of security features out of the box and HTML, CSS pages for frontend of our web application and node.js built on JavaScript.

Additionally, django provides a templating engine that allows developers to easily render HTML pages, and a web server that enables them to serve the pages to clients.. Along with these features, Bootstrap offers a set of CSS classes that can be readily used with HTML pages.

Node.js enabled us to write server-side code using JavaScript which is traditionally associated with browser based scripting. It followed an event driven non-blocking I/O model. This is because it can handle concurrent connections efficiently without blocking the execution of other tasks. It uses callbacks, promises, and async/await to asynchronous operations, allowing for scalable and high-performance applications. As suggested by professor, we used MySQL database. It communicates with the backend with a set of protocols like HTTP and SQL quiries, has good performance and reliability.

DDL File Content

```
-- Generated by Oracle SQL Developer Data Modeler 22.2.0.165.1149
--   at:        2023-04-29 12:00:18 EDT
--   site:      Oracle Database 21c
--   type:      Oracle Database 21c



-- predefined type, no DDL - MDSYS.SDO_GEOMETRY


-- predefined type, no DDL - XMLTYPE
```

```
CREATE TABLE psc_attr (
   a_id        NUMBER(10) NOT NULL,
   a_name      VARCHAR2(15) NOT NULL,
   a_des       VARCHAR2(30) NOT NULL,
   a_type      VARCHAR2(20) NOT NULL,
   a_status    VARCHAR2(10) NOT NULL,
   min_ht      NUMBER(3, 2) NOT NULL,
   a_dur       NUMBER(3) NOT NULL,
   loc_section VARCHAR2(3) NOT NULL,
   a_capacity  NUMBER(3) NOT NULL
);

COMMENT ON COLUMN psc_attr.a_id IS
   'UNIQUE ID OF ATTRACTIONS';

COMMENT ON COLUMN psc_attr.a_name IS
   'NAME OF ATTRACTIONS';

COMMENT ON COLUMN psc_attr.a_des IS
   'DESCRIPTION OF THE ATTRACTION';

COMMENT ON COLUMN psc_attr.a_type IS
   'TYPE OF THE ATTRACTION';

COMMENT ON COLUMN psc_attr.a_status IS
   'STATUS OF ATTRACTION';

COMMENT ON COLUMN psc_attr.min_ht IS
   'MINIMUM HEIGHT ALLOWED FOR ATTRACTION';
```

```
COMMENT ON COLUMN psc_attr.a_dur IS
   'DURATION OF ATTRACTION OPENED';

COMMENT ON COLUMN psc_attr.loc_section IS
   'LOCATION OF THE ATTRACTION';

COMMENT ON COLUMN psc_attr.a_capacity IS
   'CAPACITY FOR ATTRACTION';

ALTER TABLE psc_attr ADD CONSTRAINT psc_attr_pk PRIMARY KEY ( a_id );

CREATE TABLE psc_card (
   pay_id NUMBER(10) NOT NULL,
   c_name VARCHAR2(30) NOT NULL,
   c_num  VARCHAR2(15) NOT NULL,
   c_exp  DATE NOT NULL,
   c_cvv  NUMBER(3) NOT NULL,
   c_type VARCHAR2(10) NOT NULL
);

COMMENT ON COLUMN psc_card.pay_id IS
   'UNIQUE PRIMARY KEY OF PAYMENT';

COMMENT ON COLUMN psc_card.c_name IS
   'NAME ON THE CARD';

COMMENT ON COLUMN psc_card.c_num IS
   'NUMBER OF THE CARD';

COMMENT ON COLUMN psc_card.c_exp IS
   'EXPIRY DATE OF THE CARD';
```

```
COMMENT ON COLUMN psc_card.c_cvv IS
   'CVV OF THE CARD';

COMMENT ON COLUMN psc_card.c_type IS
   'TYPE OF THE CARD';

ALTER TABLE psc_card ADD CONSTRAINT psc_card_pk PRIMARY KEY ( pay_id );

CREATE TABLE psc_cat (
   ca_id   NUMBER(10) NOT NULL,
   ca_name VARCHAR2(20) NOT NULL
);

COMMENT ON COLUMN psc_cat.ca_id IS
   'UNIQUE ID OF THE CATEGORY';

COMMENT ON COLUMN psc_cat.ca_name IS
   'NAME OF THE CATEGORY';

ALTER TABLE psc_cat ADD CONSTRAINT psc_cat_pk PRIMARY KEY ( ca_id );

CREATE TABLE psc_entry (
   tic_id     NUMBER(10) NOT NULL,
   e_price    NUMBER(5, 2) NOT NULL,
   pur_method VARCHAR2(10) NOT NULL
);

COMMENT ON COLUMN psc_entry.tic_id IS
   'Unique primary key of Ticket entity';
```

```
COMMENT ON COLUMN psc_entry.e_price IS
   'PRICE OF ENTRY TICKET';

COMMENT ON COLUMN psc_entry.pur_method IS
   'METHOD OF ENTRY PURCHASE TICKET';

ALTER TABLE psc_entry ADD CONSTRAINT psc_entry_pk PRIMARY KEY ( tic_id );

CREATE TABLE psc_invoice (
   inv_id   NUMBER(10) NOT NULL,
   inv_date DATE NOT NULL,
   inv_amt  NUMBER(5, 2) NOT NULL
);

COMMENT ON COLUMN psc_invoice.inv_id IS
   'UNIQUE INVOICE ID';

COMMENT ON COLUMN psc_invoice.inv_date IS
   'DATE FOR WHICH INVOICE CREATED';

COMMENT ON COLUMN psc_invoice.inv_amt IS
   'INVOICE AMOUNT TO BE PAID';

ALTER TABLE psc_invoice ADD CONSTRAINT psc_invoice_pk PRIMARY KEY ( inv_id );

CREATE TABLE psc_menu (
   m_id    NUMBER(10) NOT NULL,
   m_item  VARCHAR2(15) NOT NULL,
   m_desc  VARCHAR2(30) NOT NULL,
   m_price NUMBER(5, 2) NOT NULL,
   str_id  NUMBER(10) NOT NULL
```

```
);

COMMENT ON COLUMN psc_menu.m_id IS
   'UNIQUE ID OF THE MENU ITEM';

COMMENT ON COLUMN psc_menu.m_item IS
   'NAME OF THE MENU ITEM';

COMMENT ON COLUMN psc_menu.m_desc IS
   'DESCRIPTION OF MENU ITEM';

COMMENT ON COLUMN psc_menu.m_price IS
   'PRICE OF EACH ITEM';

COMMENT ON COLUMN psc_menu.str_id IS
   'STORE ID FOREIGN KEY';

ALTER TABLE psc_menu ADD CONSTRAINT psc_menu_pk PRIMARY KEY ( m_id );

CREATE TABLE psc_parking (
   p_id     NUMBER(15) NOT NULL,
   p_lot    VARCHAR2(5) NOT NULL,
   spot_num NUMBER(5) NOT NULL,
   time_in  DATE NOT NULL,
   time_out DATE NOT NULL,
   p_fee    NUMBER(5, 2) NOT NULL,
   inv_id   NUMBER(10) NOT NULL
);

COMMENT ON COLUMN psc_parking.p_id IS
   'UNIQUE PARKING ID FOR VISITORS';
```

```
COMMENT ON COLUMN psc_parking.p_lot IS
   'PARKING LOT ';

COMMENT ON COLUMN psc_parking.spot_num IS
   'SPOT NUMBER OF PARKING';

COMMENT ON COLUMN psc_parking.time_in IS
   'IN TIME OF VISITORS';

COMMENT ON COLUMN psc_parking.time_out IS
   'TIME OUT OF VISITORS PARKING';

COMMENT ON COLUMN psc_parking.p_fee IS
   'PARKING FEE OF VISITORS';

COMMENT ON COLUMN psc_parking.inv_id IS
   'INVOICE ID FOREIGN KEY';

CREATE UNIQUE INDEX psc_parking__idx ON
   psc_parking (
      inv_id
   ASC );

ALTER TABLE psc_parking ADD CONSTRAINT psc_parking_pk PRIMARY KEY ( p_id );

CREATE TABLE psc_payment (
   pay_id   NUMBER(10) NOT NULL,
   pay_date DATE NOT NULL,
   pay_amt  NUMBER(6, 2) NOT NULL,
   is_card  CHAR(1) NOT NULL,
```

```
    inv_id   NUMBER(10) NOT NULL
);

ALTER TABLE psc_payment
    ADD CONSTRAINT ch_inh_psc_payment CHECK ( is_card IN ( 'N', 'Y' ) );

COMMENT ON COLUMN psc_payment.pay_id IS
    'UNIQUE PRIMARY KEY OF PAYMENT';

COMMENT ON COLUMN psc_payment.pay_date IS
    'DATE WHICH TOTAL AMOUNT IS PAID';

COMMENT ON COLUMN psc_payment.pay_amt IS
    'AMOUNT TO BE PAID';

COMMENT ON COLUMN psc_payment.is_card IS
    'IT ASKS WHETHER PAYMENT METHOD ID CARD OR CASH';

COMMENT ON COLUMN psc_payment.inv_id IS
    'FOREIGN KEY INVOICE ID';

CREATE UNIQUE INDEX psc_payment__idx ON
    psc_payment (
        inv_id
    ASC );

ALTER TABLE psc_payment ADD CONSTRAINT psc_payment_pk PRIMARY KEY ( pay_id );

CREATE TABLE psc_sh (
    sh_id      NUMBER(10) NOT NULL,
    sh_name    VARCHAR2(15) NOT NULL,
```

```
    sh_desc     VARCHAR2(30) NOT NULL,
    str_time    DATE NOT NULL,
    end_time    DATE NOT NULL,
    wh_acc      CHAR(1) NOT NULL,
    sh_price    NUMBER(5, 2) NOT NULL,
    sh_type_id NUMBER(10) NOT NULL
);

COMMENT ON COLUMN psc_sh.sh_id IS
    'UNIQUE ID OF THE SHOW';

COMMENT ON COLUMN psc_sh.sh_name IS
    'NAME OF THE SHOW';

COMMENT ON COLUMN psc_sh.sh_desc IS
    'DESCRIPTION OF THE SHOW';

COMMENT ON COLUMN psc_sh.str_time IS
    'START TIME OF THE SHOW';

COMMENT ON COLUMN psc_sh.end_time IS
    'END TIME OF THE SHOW';

COMMENT ON COLUMN psc_sh.wh_acc IS
    'WHEEL CHAIR ACCESSIBLE OR NOT';

COMMENT ON COLUMN psc_sh.sh_price IS
    'PRICE OF THE SHOW';

COMMENT ON COLUMN psc_sh.sh_type_id IS
    'SH_TYPE_ID FOREIGN KEY ';
```

```
ALTER TABLE psc_sh ADD CONSTRAINT psc_sh_pk PRIMARY KEY ( sh_id );

CREATE TABLE psc_show (
   tic_id NUMBER(10) NOT NULL,
   sh_id  NUMBER(10) NOT NULL
);

COMMENT ON COLUMN psc_show.tic_id IS
   'Unique primary key of Ticket entity';

COMMENT ON COLUMN psc_show.sh_id IS
   'SH_ID FOREIGN KEY FOR SHOW';

ALTER TABLE psc_show ADD CONSTRAINT psc_show_pk PRIMARY KEY ( tic_id );

CREATE TABLE psc_store (
   str_id   NUMBER(10) NOT NULL,
   str_name VARCHAR2(20) NOT NULL,
   ca_id    NUMBER(10) NOT NULL
);

COMMENT ON COLUMN psc_store.str_id IS
   'UNIQUE ID OF THE STORE';

COMMENT ON COLUMN psc_store.str_name IS
   'STORE NAME';

COMMENT ON COLUMN psc_store.ca_id IS
   'CATEGORY ID FOREIGN KEY';
```

```
ALTER TABLE psc_store ADD CONSTRAINT psc_store_pk PRIMARY KEY ( str_id );

CREATE TABLE psc_str_ord (
   order_id NUMBER(10) NOT NULL,
   o_qua    NUMBER(5) NOT NULL,
   o_price  NUMBER(5, 2) NOT NULL,
   v_id     NUMBER(10) NOT NULL,
   str_id   NUMBER(10) NOT NULL,
   inv_id   NUMBER(10) NOT NULL
);

COMMENT ON COLUMN psc_str_ord.order_id IS
   'UNIQUE ID OF PRIMARY KEY';

COMMENT ON COLUMN psc_str_ord.o_qua IS
   'QUANTITY OF STORE ORDERS';

COMMENT ON COLUMN psc_str_ord.o_price IS
   'UNIT_PRICE OF PRODUCT';

COMMENT ON COLUMN psc_str_ord.v_id IS
   'VISITOR ID FOREIGN KEY';

COMMENT ON COLUMN psc_str_ord.str_id IS
   'STORE ID FOREIGN KEY';

COMMENT ON COLUMN psc_str_ord.inv_id IS
   'INVOICE ID FOREIGN KEY';

ALTER TABLE psc_str_ord ADD CONSTRAINT psc_str_ord_pk PRIMARY KEY ( order_id );
```

```
CREATE TABLE psc_stype (
   sh_type_id NUMBER(10) NOT NULL,
   type_name  VARCHAR2(10) NOT NULL
);

COMMENT ON COLUMN psc_stype.sh_type_id IS
   'UNIQUE ID OF SHOW TYPE';

COMMENT ON COLUMN psc_stype.type_name IS
   'SHOW TYPE NAME';

ALTER TABLE psc_stype ADD CONSTRAINT psc_stype_pk PRIMARY KEY ( sh_type_id );

CREATE TABLE psc_summary (
   visitor_id   NUMBER(10) NOT NULL,
   visitor_date DATE NOT NULL,
   tick_id      NUMBER(10) NOT NULL,
   e_amt        NUMBER(5, 2) NOT NULL,
   sho_id       NUMBER(10) NOT NULL,
   sho_amt      NUMBER(5, 2) NOT NULL,
   par_id       NUMBER(10) NOT NULL,
   par_amt      NUMBER(5, 2) NOT NULL,
   sto_id       NUMBER(10) NOT NULL,
   sto_amt      NUMBER(5, 2) NOT NULL
);

COMMENT ON COLUMN psc_summary.visitor_id IS
   'UNIQUE PRIMARY KEY OF SUMMARY TABLE';

COMMENT ON COLUMN psc_summary.visitor_date IS
   'VISITOR DATE PRIMARY KEY';
```

```
COMMENT ON COLUMN psc_summary.tick_id IS
   'TICKET ID OF VISITOR';

COMMENT ON COLUMN psc_summary.e_amt IS
   'AMOUNT OF THE TICKET';

COMMENT ON COLUMN psc_summary.sho_id IS
   'SHOW ID OF VISITOR';

COMMENT ON COLUMN psc_summary.sho_amt IS
   'SHOW AMOUNT';

COMMENT ON COLUMN psc_summary.par_id IS
   'ID OF PARKING';

COMMENT ON COLUMN psc_summary.par_amt IS
   'PARKING AMT';

COMMENT ON COLUMN psc_summary.sto_id IS
   'STORE ID';

COMMENT ON COLUMN psc_summary.sto_amt IS
   'STORE AMOUNT';

ALTER TABLE psc_summary ADD CONSTRAINT psc_summary_pk PRIMARY KEY ( visitor_id,
                                                                    visitor_date );

CREATE TABLE psc_tic (
   tic_id     NUMBER(10) NOT NULL,
   t_category CHAR(1) NOT NULL,
```

```
    t_method   VARCHAR2(10) NOT NULL,
    p_dt       DATE NOT NULL,
    v_dt       DATE NOT NULL,
    t_type     VARCHAR2(10) NOT NULL,
    t_discount NUMBER(3, 2),
    v_id       NUMBER(10) NOT NULL,
    inv_id     NUMBER(10) NOT NULL
);

ALTER TABLE psc_tic
    ADD CONSTRAINT ch_inh_psc_tic CHECK ( t_category IN ( 'E', 'S' ) );

COMMENT ON COLUMN psc_tic.tic_id IS
    'Unique primary key of Ticket entity';

COMMENT ON COLUMN psc_tic.t_category IS
    'T_CATEGORY, ENTRY OR SHOW TICKET';

COMMENT ON COLUMN psc_tic.t_method IS
    'Ticket Method whether online or onsite';

COMMENT ON COLUMN psc_tic.p_dt IS
    'Purchase date of the ticket';

COMMENT ON COLUMN psc_tic.v_dt IS
    'Visit date of the visitor';

COMMENT ON COLUMN psc_tic.t_type IS
    'Type of the ticket';

COMMENT ON COLUMN psc_tic.t_discount IS
```

```
    'Discount of the ticket';

COMMENT ON COLUMN psc_tic.v_id IS
    'VISITOR ID FOREIGN KEY';

COMMENT ON COLUMN psc_tic.inv_id IS
    'INVOICE ID FOREIGN KEY';

ALTER TABLE psc_tic ADD CONSTRAINT psc_tic_pk PRIMARY KEY ( tic_id );

CREATE TABLE psc_tic_at (
    a_date DATE NOT NULL,
    tic_id NUMBER(10) NOT NULL,
    a_id   NUMBER(10) NOT NULL
);

COMMENT ON COLUMN psc_tic_at.a_date IS
    'DATE WHICH ATTRACTION IS VISITED';

COMMENT ON COLUMN psc_tic_at.tic_id IS
    'TIC_ID FOREIGN KEY';

COMMENT ON COLUMN psc_tic_at.a_id IS
    'A_ID FOREIGN KEY';

ALTER TABLE psc_tic_at ADD CONSTRAINT psc_tic_at_pk PRIMARY KEY ( a_date );

CREATE TABLE psc_visitors (
    v_id      NUMBER(10) NOT NULL,
    v_fname   VARCHAR2(15) NOT NULL,
    v_lname   VARCHAR2(15) NOT NULL,
```

```
    v_dob     DATE NOT NULL,
    v_email   VARCHAR2(30) NOT NULL,
    v_pho     NUMBER(10) NOT NULL,
    v_street  VARCHAR2(15) NOT NULL,
    v_city    VARCHAR2(15) NOT NULL,
    v_state   VARCHAR2(15) NOT NULL,
    v_zipcode VARCHAR2(12) NOT NULL,
    v_country VARCHAR2(15) NOT NULL,
    v_type    VARCHAR2(10) NOT NULL,
    p_id      NUMBER(15) NOT NULL
);

COMMENT ON COLUMN psc_visitors.v_id IS
    'Unique visitor ID';

COMMENT ON COLUMN psc_visitors.v_fname IS
    'FIRST NAME OF THE VISITOR';

COMMENT ON COLUMN psc_visitors.v_lname IS
    'LAST NAME OF THE VISITOR';

COMMENT ON COLUMN psc_visitors.v_dob IS
    'Date of Birth of visitor';

COMMENT ON COLUMN psc_visitors.v_email IS
    'EMAIL ID OF THE VISITOR';

COMMENT ON COLUMN psc_visitors.v_pho IS
    'PHONE NUMBER OF THE VISITOR';

COMMENT ON COLUMN psc_visitors.v_street IS
```

```
    'STREET ADDRESS OF THE VISITOR';

COMMENT ON COLUMN psc_visitors.v_city IS
    'CITY ADDRESS OF VISITOR';

COMMENT ON COLUMN psc_visitors.v_state IS
    'STATE ADDRESS OF VISITOR';

COMMENT ON COLUMN psc_visitors.v_zipcode IS
    'ZIPCODE ADDRESS OF VISITOR';

COMMENT ON COLUMN psc_visitors.v_country IS
    'COUNTRY OF VISITOR';

COMMENT ON COLUMN psc_visitors.v_type IS
    'VISITOR TYPE';

COMMENT ON COLUMN psc_visitors.p_id IS
    'PARKING ID FOREIGN KEY';

ALTER TABLE psc_visitors ADD CONSTRAINT psc_visitors_pk PRIMARY KEY ( v_id );

ALTER TABLE psc_card
    ADD CONSTRAINT psc_card_psc_payment_fk FOREIGN KEY ( pay_id )
        REFERENCES psc_payment ( pay_id );

ALTER TABLE psc_entry
    ADD CONSTRAINT psc_entry_psc_tic_fk FOREIGN KEY ( tic_id )
        REFERENCES psc_tic ( tic_id );

ALTER TABLE psc_menu
```

```
   ADD CONSTRAINT psc_menu_psc_store_fk FOREIGN KEY ( str_id )
      REFERENCES psc_store ( str_id );

ALTER TABLE psc_parking
   ADD CONSTRAINT psc_parking_psc_invoice_fk FOREIGN KEY ( inv_id )
      REFERENCES psc_invoice ( inv_id );

ALTER TABLE psc_payment
   ADD CONSTRAINT psc_payment_psc_invoice_fk FOREIGN KEY ( inv_id )
      REFERENCES psc_invoice ( inv_id );

ALTER TABLE psc_sh
   ADD CONSTRAINT psc_sh_psc_stype_fk FOREIGN KEY ( sh_type_id )
      REFERENCES psc_stype ( sh_type_id );

ALTER TABLE psc_show
   ADD CONSTRAINT psc_show_psc_sh_fk FOREIGN KEY ( sh_id )
      REFERENCES psc_sh ( sh_id );

ALTER TABLE psc_show
   ADD CONSTRAINT psc_show_psc_tic_fk FOREIGN KEY ( tic_id )
      REFERENCES psc_tic ( tic_id );

ALTER TABLE psc_store
   ADD CONSTRAINT psc_store_psc_cat_fk FOREIGN KEY ( ca_id )
      REFERENCES psc_cat ( ca_id );

ALTER TABLE psc_str_ord
   ADD CONSTRAINT psc_str_ord_psc_invoice_fk FOREIGN KEY ( inv_id )
      REFERENCES psc_invoice ( inv_id );
```

```
ALTER TABLE psc_str_ord
   ADD CONSTRAINT psc_str_ord_psc_store_fk FOREIGN KEY ( str_id )
      REFERENCES psc_store ( str_id );

ALTER TABLE psc_str_ord
   ADD CONSTRAINT psc_str_ord_psc_visitors_fk FOREIGN KEY ( v_id )
      REFERENCES psc_visitors ( v_id );

ALTER TABLE psc_tic_at
   ADD CONSTRAINT psc_tic_at_psc_attr_fk FOREIGN KEY ( a_id )
      REFERENCES psc_attr ( a_id );

ALTER TABLE psc_tic_at
   ADD CONSTRAINT psc_tic_at_psc_tic_fk FOREIGN KEY ( tic_id )
      REFERENCES psc_tic ( tic_id );

ALTER TABLE psc_tic
   ADD CONSTRAINT psc_tic_psc_invoice_fk FOREIGN KEY ( inv_id )
      REFERENCES psc_invoice ( inv_id );

ALTER TABLE psc_tic
   ADD CONSTRAINT psc_tic_psc_visitors_fk FOREIGN KEY ( v_id )
      REFERENCES psc_visitors ( v_id );

ALTER TABLE psc_visitors
   ADD CONSTRAINT psc_visitors_psc_parking_fk FOREIGN KEY ( p_id )
      REFERENCES psc_parking ( p_id );

CREATE OR REPLACE TRIGGER arc_fkarc_2_psc_entry BEFORE
   INSERT OR UPDATE OF tic_id ON psc_entry
   FOR EACH ROW
```

```
DECLARE
   d CHAR(1);
BEGIN
   SELECT
      a.t_category
   INTO d
   FROM
      psc_tic a
   WHERE
      a.tic_id = :new.tic_id;

   IF ( d IS NULL OR d <> 'E' ) THEN
      raise_application_error(-20223, 'FK PSC_ENTRY_PSC_TIC_FK in Table PSC_ENTRY violates
Arc constraint on Table PSC_TIC - discriminator column T_CATEGORY doesn"t have value "E"'
      );
   END IF;

EXCEPTION
   WHEN no_data_found THEN
      NULL;
   WHEN OTHERS THEN
      RAISE;
END;
/

CREATE OR REPLACE TRIGGER arc_fkarc_2_psc_show BEFORE
   INSERT OR UPDATE OF tic_id ON psc_show
   FOR EACH ROW
DECLARE
   d CHAR(1);
BEGIN
```

```
    SELECT
        a.t_category
    INTO d
    FROM
        psc_tic a
    WHERE
        a.tic_id = :new.tic_id;

    IF ( d IS NULL OR d <> 'S' ) THEN
        raise_application_error(-20223, 'FK PSC_SHOW_PSC_TIC_FK in Table PSC_SHOW violates Arc constraint on Table PSC_TIC - discriminator column T_CATEGORY doesn"t have value "S"'
        );
    END IF;

EXCEPTION
    WHEN no_data_found THEN
        NULL;
    WHEN OTHERS THEN
        RAISE;
END;
/
```

List of Tables and their corresponding number of records:

```
SELECT table_name, table_rows
FROM information_schema.tables
WHERE table_schema = 'finalproj1';
```

table_name	table_rows
psc_attr	15
psc_card	10
psc_cat	5
psc_entry	18
psc_invoice	14
psc_menu	20
psc_parking	14
psc_payment	14
psc_sh	10
psc_show	7
psc_store	16
psc_str_ord	30
psc_stype	5
psc_summary	10
psc_tic	36
psc_tic_at	10
psc_visitors	20

Screenshots:

Signup

Username

Password

Password confirmation

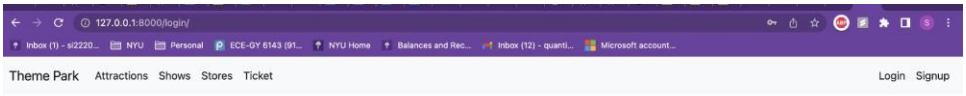
Phone

Identity type ---------

Identity number

sign up

Already have an account? Login



Theme Park Attractions Shows Stores Ticket Login Signup

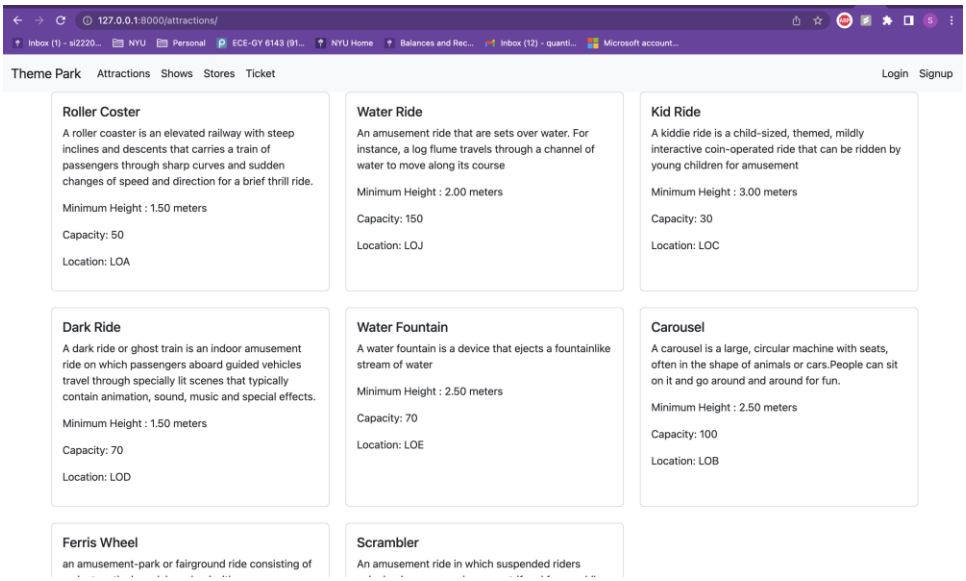
Login

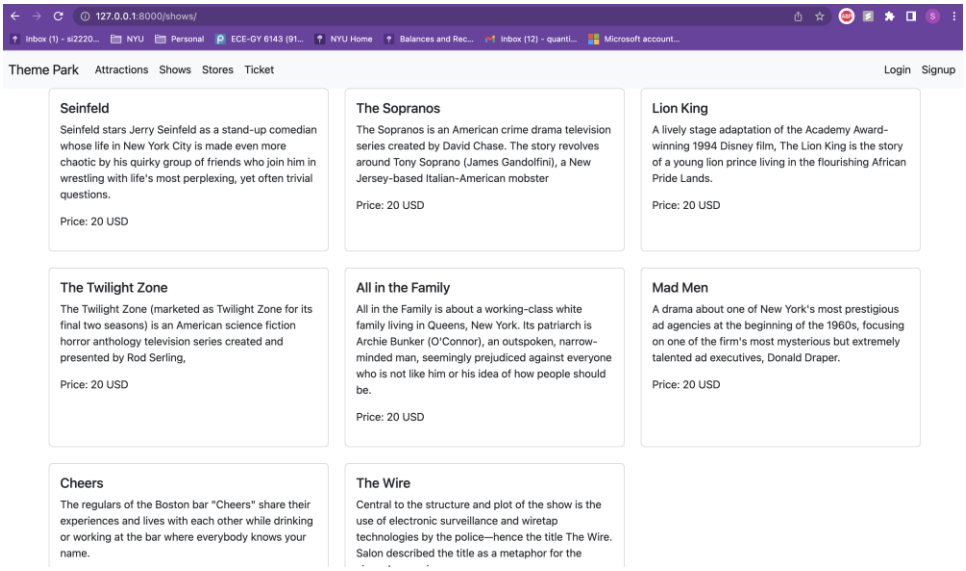
Username

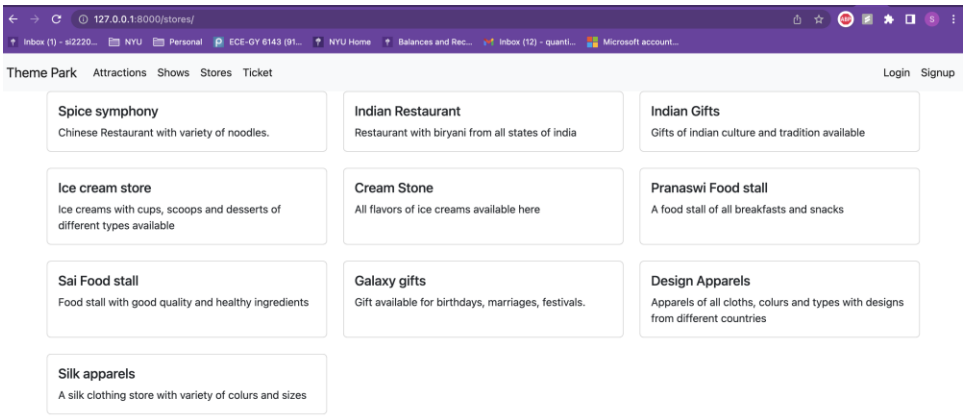
Password

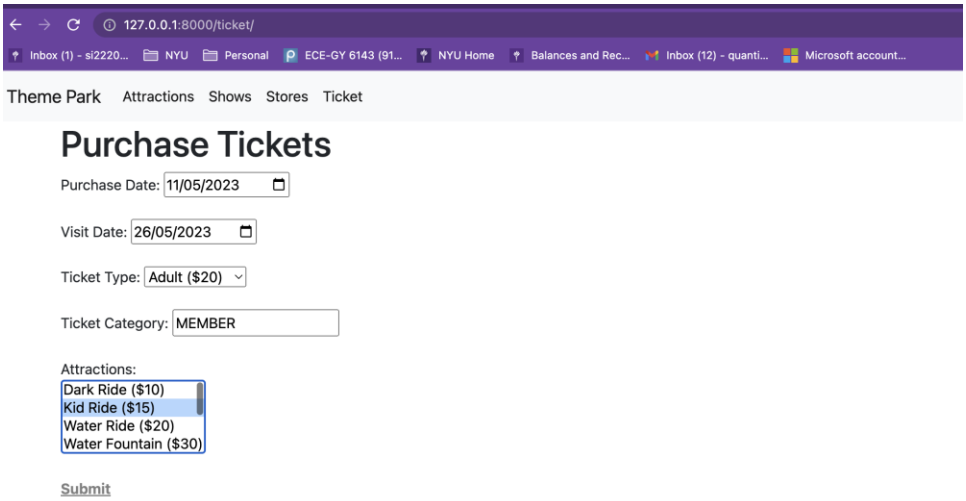
Login

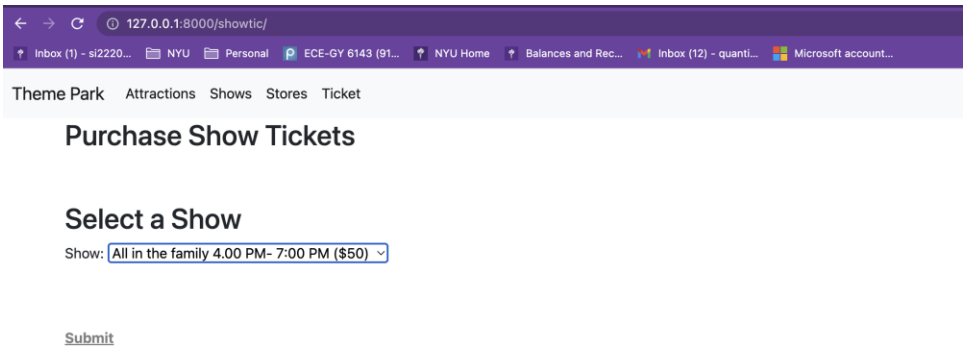
Don't have an account? Sign Up


127.0.0.1:8000/attractions/
Theme Park
Attractions
Shows
Stores
Ticket
Login
Signup
Roller Coster
A roller coaster is an elevated railway with steep inclines and descents that carries a train of passengers through sharp curves and sudden changes of speed and direction for a brief thrill ride.
Minimum Height : 1.50 meters
Capacity: 50
Location: LOA
Water Ride
An amusement ride that are sets over water. For instance, a log flume travels through a channel of water to move along its course
Minimum Height : 2.00 meters
Capacity: 150
Location: LOJ
Kid Ride
A kiddie ride is a child-sized, themed, mildly interactive coin-operated ride that can be ridden by young children for amusement
Minimum Height : 3.00 meters
Capacity: 30
Location: LOC
Dark Ride
A dark ride or ghost train is an indoor amusement ride on which passengers aboard guided vehicles travel through specially lit scenes that typically contain animation, sound, music and special effects.
Minimum Height : 1.50 meters
Capacity: 70
Location: LOD
Water Fountain
A water fountain is a device that ejects a fountainlike stream of water
Minimum Height : 2.50 meters
Capacity: 70
Location: LOE
Carousel
A carousel is a large, circular machine with seats, often in the shape of animals or cars.People can sit on it and go around and around for fun.
Minimum Height : 2.50 meters
Capacity: 100
Location: LOB
Ferris Wheel
an amusement-park or fairground ride consisting of
Scrambler
An amusement ride in which suspended riders


127.0.0.1:8000/shows/
Theme Park
Attractions
Shows
Stores
Ticket
Login
Signup
Seinfeld
Seinfeld stars Jerry Seinfeld as a stand-up comedian whose life in New York City is made even more chaotic by his quirky group of friends who join him in wrestling with life's most perplexing, yet often trivial questions.
Price: 20 USD
The Sopranos
The Sopranos is an American crime drama television series created by David Chase. The story revolves around Tony Soprano (James Gandolfini), a New Jersey-based Italian-American mobster
Price: 20 USD
Lion King
A lively stage adaptation of the Academy Award-winning 1994 Disney film, The Lion King is the story of a young lion prince living in the flourishing African Pride Lands.
Price: 20 USD
The Twilight Zone
The Twilight Zone (marketed as Twilight Zone for its final two seasons) is an American science fiction horror anthology television series created and presented by Rod Serling,
Price: 20 USD
All in the Family
All in the Family is about a working-class white family living in Queens, New York. Its patriarch is Archie Bunker (O'Connor), an outspoken, narrow-minded man, seemingly prejudiced against everyone who is not like him or his idea of how people should be.
Price: 20 USD
Mad Men
A drama about one of New York's most prestigious ad agencies at the beginning of the 1960s, focusing on one of the firm's most mysterious but extremely talented ad executives, Donald Draper.
Price: 20 USD
Cheers
The regulars of the Boston bar "Cheers" share their experiences and lives with each other while drinking or working at the bar where everybody knows your name.
The Wire
Central to the structure and plot of the show is the use of electronic surveillance and wiretap technologies by the police—hence the title The Wire. Salon described the title as a metaphor for the


127.0.0.1:8000/stores/
Theme Park
Attractions
Shows
Stores
Ticket
Login
Signup
Spice symphony
Chinese Restaurant with variety of noodles.
Indian Restaurant
Restaurant with biryani from all states of india
Indian Gifts
Gifts of indian culture and tradition available
Ice cream store
Ice creams with cups, scoops and desserts of different types available
Cream Stone
All flavors of ice creams available here
Pranaswi Food stall
A food stall of all breakfasts and snacks
Sai Food stall
Food stall with good quality and healthy ingredients
Galaxy gifts
Gift available for birthdays, marriages, festivals.
Design Apparels
Apparels of all cloths, colurs and types with designs from different countries
Silk apparels
A silk clothing store with variety of colurs and sizes


127.0.0.1:8000/ticket/
Theme Park
Attractions
Shows
Stores
Ticket
Purchase Tickets
Purchase Date: 11/05/2023
Visit Date: 26/05/2023
Ticket Type: Adult ($20)
Ticket Category: MEMBER
Attractions:
Dark Ride ($10)
Kid Ride ($15)
Water Ride ($20)
Water Fountain ($30)
Submit


127.0.0.1:8000/showtic/
Inbox (1) - si2220...
NYU
Personal
ECE-GY 6143 (91...
NYU Home
Balances and Rec...
Inbox (12) - quanti...
Microsoft account...
Theme Park
Attractions
Shows
Stores
Ticket
Purchase Show Tickets
Select a Show
Show:
All in the family 4.00 PM- 7:00 PM ($50)
Submit

127.0.0.1:8000/storetic/

Inbox (1) - si2220... NYU Personal ECE-GY 6143 (91... NYU Home Balances and Rec... Inbox (12) - quanti... Microsoft account...

Theme Park Attractions Shows Stores Ticket

Buy our products

Spice Symphony

- Biryani ($10)
- Coke ($15)
- Spicy Noodles ($20)
- Alu Parota ($30)

Indian Restaurant

- Full Meals ($10)
- Sprite ($15)
- Chicken Noodles ($20)
- Gobi Parota ($30)

Indian Gifts

- Barbie doll ($10)
- Spider man ($15)
- Scenary ($20)
- Photo frame ($30)

Ice Cream Store

- Chocolate icecream ($10)
- Strawbwrry flavor ($15)
- Vanilla Flavor ($20)
- Butterscotch flavor ($30)

Pranaswi Food Stall

- Dosa ($10)
- IDLI ($15)
- Vada ($20)
- Puri ($30)

Ice Cream Store

- Chocolate icecream ($10)
- Strawbwrry flavor ($15)
- Vanilla Flavor ($20)
- Butterscotch flavor ($30)

Galaxy Gifts

- Bat Man doll ($10)
- Spider man ($15)
- Scenary ($20)
- Photo frame ($30)

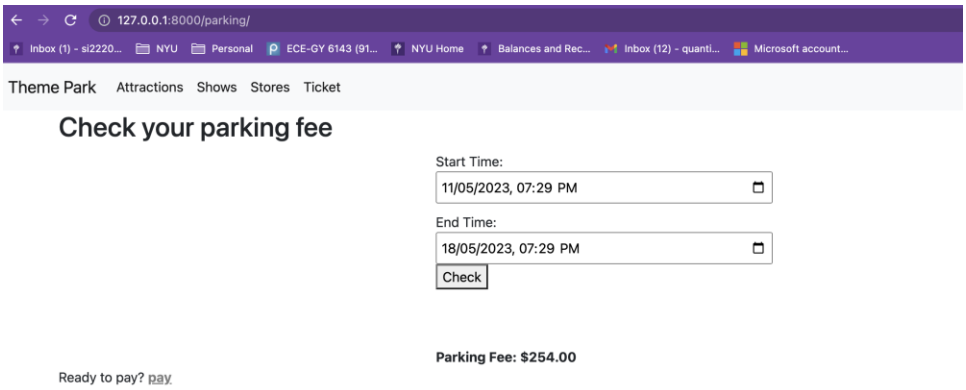
Design Apparels

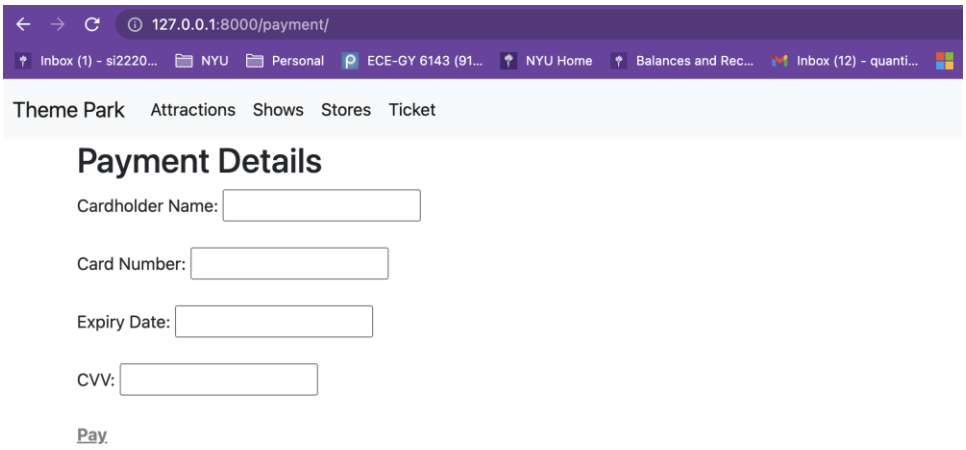
- Flower design ($10)
- Spider man ($15)
- Scenary ($20)
- Photo frame ($30)

Silk Apparels

- Red Silk cloth ($10)
- Spider man ($15)
- Scenary ($20)
- Photo frame ($30)

Submit


127.0.0.1:8000/parking/
Inbox (1) - si2220...
NYU
Personal
ECE-GY 6143 (91...
NYU Home
Balances and Rec...
Inbox (12) - quanti...
Microsoft account...
Theme Park
Attractions
Shows
Stores
Ticket
Check your parking fee
Start Time:
11/05/2023, 07:29 PM
End Time:
18/05/2023, 07:29 PM
Check
Parking Fee: $254.00
Ready to pay? pay


127.0.0.1:8000/payment/
Inbox (1) - si2220...
NYU
Personal
ECE-GY 6143 (91...
NYU Home
Balances and Rec...
Inbox (12) - quanti...
Theme Park
Attractions
Shows
Stores
Ticket
Payment Details
Cardholder Name:
Card Number:
Expiry Date:
CVV:
Pay

127.0.0.1:8000/payment/

Inbox (1) - si2220... | NYU | Personal | ECE-GY 6143 (91... | NYU Home | Balances and Rec...

Theme Park Attractions Shows Stores Ticket

Payment Details

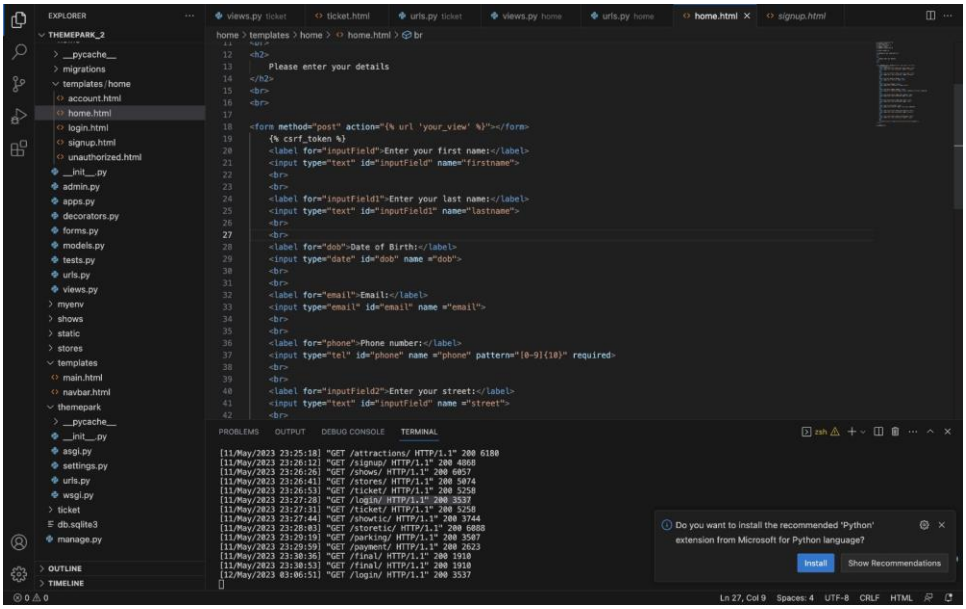
Cardholder Name: sai

Card Number: 1111 1111 1111 1111

Expiry Date: 12/2023

CVV: 123

Pay



Security Features

SQL Injection

SQL Injection is a critical web-based application security vulnerability that can have severe consequences if not addressed properly. It occurs when attackers exploit weaknesses in an application's database by taking advantage of insufficient input validation and to inject malicious SQL code. Thus, leading to unauthorized access, data manipulation, and potentially even full control over the database.

The impact of SQL Injection can be devastating. Attackers can bypass authentication mechanisms, gain access to sensitive information, modify or delete data, and even execute arbitrary commands on the database server. This poses a significant threat to the confidentiality, integrity, and availability of data.

Django provides protection against SQL injection attacks by using query parameterization when constructing querysets. This means that query parameters are passed as separate arguments to the database backend and are escaped by the underlying database driver. By avoiding the embedding of query parameters in the query string, the database backend is not vulnerable to SQL injection attacks that exploit poorly constructed queries.

Transaction concurrency

Transaction concurrency is the ability of a database management system to handle multiple transactions that occur simultaneously that can cause inconsistencies or conflicts. Protection is needed against Transaction concurrency to prevent inconsistencies or conflicts that may arise when multiple transactions are executed simultaneously. The database management system must use various techniques such as locking, multiversion concurrency control, optimistic concurrency control, and snapshot isolation to ensure that transactions are executed in a coordinated manner while maintaining data consistency, reliability, and integrity. Without proper protection against transaction concurrency, inconsistencies and conflicts can lead to data corruption, loss, or other unintended consequences that can negatively impact the functionality and performance of the application. Effective transaction concurrency protection is crucial for ensuring the efficient and reliable operation of database systems in high-traffic applications.

Transaction concurrency protection in Node.js can be achieved through techniques such as locking, optimistic concurrency control, and multiversion concurrency control. These techniques help prevent inconsistencies and conflicts that may arise when multiple transactions are executed simultaneously. We implemented these techniques and ensured data consistency, reliability, and integrity in high-traffic conditions.

Deadlock

Deadlock is a situation in a multi-process system where two or more processes are blocked and unable to proceed because each is waiting for one of the others to complete or release a resource. In other words, each process is stuck waiting for a resource that is being held by another process, which in turn is waiting for a resource that the first process is holding. Deadlocks can cause the entire system to freeze and can only be resolved by aborting one or more of the processes involved.

To prevent deadlocks in Node.js, there are techniques that can be employed, such as timeouts, polling, and async/await functions. Our approach was to set a timeout for the database operation, and if it takes longer than the specified time, roll back the transaction to avoid a deadlock.

Lesson Learned

In the technical point of view, we learnt how to create a website right from the scratch. We are able to manage or change the Django project according to our requirements. We tried so hard to connect the front-end localhost with backend node.js server-side host. This is the point where we faced a small issue. We got to know that Django also has an in-built database that stores user given input.

Being the novice students to this course, we learned everything from the basics. This project has made us an experienced person in databases and their applications. Now, we have a clear idea of how data entered into a web application will be stored and managed in the database and how developers interact with the database. Though it is a challenging project, it is interesting, and we had a great learning of databases, frontend and backend web applications. Moreover, our coordination skills to work in a team and presentation skills have improved. However, time management became our major constraint because we have different courses and different schedules.

Business Analysis with Project Data

Q1) Table joins with at least 3 tables in join

Business information to retrieve:

For each ticket, display all the attributes associated with the ticket and also display the show's ticket associated with it.

```
SELECT *
FROM psc_tic
JOIN psc_tic_at ON psc_tic.tic_id = psc_tic_at.tic_id
JOIN psc_show ON psc_tic_at.tic_id = psc_show.tic_id;
```

100% 54:31

Result Grid Filter Rows: Search Export:

tic_id	t_category	t_method	p_dt	v_dt	t_type	t_discount	v_id	inv_id	a_date	tic_id	a_id	tic_id	psc_sh_sh_id	s_discount
1008	S	online	2023-03-29 00:00:00	2023-03-29 00:00:00	ADULT	0.10	4	713	2023-01-06 00:00:00	1008	507	1008	903	4.00
1006	S	online	2023-03-29 00:00:00	2023-03-29 00:00:00	CHILD	0.05	3	712	2023-02-02 00:00:00	1006	505	1006	902	3.00
1004	S	online	2023-03-01 00:00:00	2023-03-03 00:00:00	ADULT	0.00	2	711	2023-03-02 00:00:00	1004	505	1004	901	2.00
1010	S	offline	2023-04-02 00:00:00	2023-04-02 00:00:00	MEMBER	0.20	5	714	2023-03-21 00:00:00	1010	509	1010	904	5.00
1002	S	online	2023-03-21 00:00:00	2023-04-01 00:00:00	ADULT	0.10	1	710	2023-04-01 00:00:00	1002	502	1002	900	1.00

Result Grid Form Editor Field Types

Result 9 Read Only

Q2) Multi-row subquery

Business information to retrieve: Display all the ticket that have method of purchase as online

```
SELECT * FROM psc_tic WHERE tic_id in (
SELECT tic_id FROM psc_tic WHERE t_method =  'online'
);
```

00% 3:35

Result Grid Filter Rows: Search Edit: Export/Import:

tic_id	t_category	t_method	p_dt	v_dt	t_type	t_discount	v_id	inv_id
1001	E	online	2023-03-01 00:00:00	2023-04-01 00:00:00	ADULT	0.10	1	710
1002	S	online	2023-03-21 00:00:00	2023-04-01 00:00:00	ADULT	0.10	1	710
1004	S	online	2023-03-01 00:00:00	2023-03-03 00:00:00	ADULT	0.00	2	711
1005	E	online	2023-03-29 00:00:00	2023-03-29 00:00:00	CHILD	0.00	3	712
1006	S	online	2023-03-29 00:00:00	2023-03-29 00:00:00	CHILD	0.05	3	712
1007	E	online	2023-03-29 00:00:00	2023-03-29 00:00:00	ADULT	0.05	4	713
1008	S	online	2023-03-29 00:00:00	2023-03-29 00:00:00	ADULT	0.10	4	713
1009	E	online	2023-03-29 00:00:00	2023-03-29 00:00:00	SENIOR	0.15	5	714
1011	E	online	2023-03-19 00:00:00	2023-02-24 00:00:00	SENIOR	0.25	6	711
1012	S	online	2023-01-05 00:00:00	2023-01-15 00:00:00	SENIOR	0.30	6	712
1013	E	online	2023-03-01 00:00:00	2023-04-01 00:00:00	ADULT	0.10	7	711
1014	S	online	2023-03-21 00:00:00	2023-04-01 00:00:00	ADULT	0.20	8	710
1016	S	online	2023-03-01 00:00:00	2023-03-03 00:00:00	ADULT	0.10	10	710
1017	E	online	2023-03-29 00:00:00	2023-03-29 00:00:00	CHILD	0.15	11	710
1018	S	online	2023-03-29 00:00:00	2023-03-29 00:00:00	CHILD	0.10	12	710
1019	E	online	2023-03-29 00:00:00	2023-03-29 00:00:00	ADULT	0.00	1	710
1020	S	online	2023-03-29 00:00:00	2023-03-29 00:00:00	ADULT	0.00	13	710
1021	E	online	2023-03-29 00:00:00	2023-03-29 00:00:00	SENIOR	0.00	1	710
1023	E	online	2023-03-19 00:00:00	2023-02-24 00:00:00	SENIOR	0.05	5	710
1024	S	online	2023-01-05 00:00:00	2023-01-15 00:00:00	SENIOR	0.10	6	710
1025	E	online	2023-03-01 00:00:00	2023-04-01 00:00:00	ADULT	0.15	1	710
1026	S	online	2023-03-21 00:00:00	2023-04-01 00:00:00	ADULT	0.10	7	710
1028	S	online	2023-03-01 00:00:00	2023-03-03 00:00:00	ADULT	0.10	9	713
1029	E	online	2023-03-29 00:00:00	2023-03-29 00:00:00	CHILD	0.00	10	714
1030	S	online	2023-03-29 00:00:00	2023-03-29 00:00:00	CHILD	0.00	11	714
1031	E	online	2023-03-29 00:00:00	2023-03-29 00:00:00	ADULT	0.00	12	711
1032	S	online	2023-03-29 00:00:00	2023-03-29 00:00:00	ADULT	0.00	13	712
1033	E	online	2023-03-29 00:00:00	2023-03-29 00:00:00	SENIOR	0.00	14	712
1035	E	online	2023-03-19 00:00:00	2023-02-24 00:00:00	SENIOR	0.00	1	710
1036	S	online	2023-01-05 00:00:00	2023-01-15 00:00:00	SENIOR	0.00	2	710
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Result Grid Form Editor Field Types Query Stats Execution Plan

Q3) Correlated subquery

Business information to retrieve: This query finds tickets whose expense is lower than the average expense from entry.

```
SELECT TIC_ID, E_PRICE
FROM PSC_ENTRY A JOIN PSC_SUMMARY B ON A.TIC_ID = B.TICK_ID
WHERE E_PRICE < (SELECT AVG(E_PRICE) FROM PSC_ENTRY) ;
```

00% 1:37

Result Grid Filter Rows: Search Export:

TIC_ID	E_PRICE
1001	20.00

Q4) SET operator query

Business information to retrieve: Find and display the ID, date be it visit or purchase, that have been later than March 1st, 2023

```sql
SELECT VISITOR_ID, VISITOR_DATE FROM PSC_SUMMARY
WHERE VISITOR_DATE BETWEEN DATE('2023-03-01') AND CURDATE()
UNION
SELECT PAY_ID, PAY_DATE FROM PSC_PAYMENT
WHERE PAY_DATE BETWEEN DATE('2023-03-01') AND CURDATE();
```

100% 57:61

Result Grid Filter Rows: Search Export:

VISITOR_ID	VISITOR_DATE
1	2023-03-01 00:00:00
2	2023-03-01 00:00:00
3	2023-03-01 00:00:00
4	2023-03-01 00:00:00
5	2023-03-01 00:00:00
6	2023-03-01 00:00:00
7	2023-03-01 00:00:00
8	2023-03-01 00:00:00
9	2023-03-01 00:00:00
10	2023-03-01 00:00:00
301	2023-04-01 10:00:00
302	2023-04-02 10:00:00
303	2023-04-03 10:00:00
304	2023-04-04 10:00:00
305	2023-03-01 10:00:00
306	2023-03-02 10:00:00
307	2023-03-03 10:00:00
311	2023-03-03 10:00:00

Q5) Query with inline view or WITH clause

Business information to retrieve: The query finds and displays the total amount of tickets bought that have purchase method as 'Cash'.

```sql
SELECT SUM(E_PRICE) FROM (
SELECT * FROM PSC_ENTRY ORDER BY E_PRICE DESC) AS PRICE
WHERE PUR_METHOD = "CASH";
```

100% 27:48

Result Grid Filter Rows: Search Export:

SUM(E_PRICE)
75.00

Q6) TOP-N query

Business information to retrieve:

The query finds the top 3 “members” who have made highest entry payments and displays ID, First Name, Last Name, Amount.

```
SELECT PSC_SUMMARY.VISITOR_ID, PSC_SUMMARY.E_AMT, V_FNAME, V_LNAME
FROM PSC_VISITORS
JOIN PSC_SUMMARY ON PSC_SUMMARY.VISITOR_ID = PSC_VISITORS.V_ID
WHERE V_TYPE = 'MEMBER' ORDER BY V_FNAME DESC
LIMIT 3;
```

100% 19:52

Result Grid Filter Rows: Search Export:

VISITOR_ID	E_AMT	V_FNAME	V_LNAME
10	30.00	CHANDANA	KASAM
3	30.00	AMULYA	GUNREDDY