



AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)

FACULTY OF SCIENCE & TECHNOLOGY **DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

Summer 2024-2025

Section: B , Group: 6

PROJECT ON

International Trade Fair Management System

Supervised By

Juena Ahmed Noshin

Submitted By

| Name | ID | Contribution |
|---------------------------------|------------|--|
| 1.Soumik Sarker | 22-46929-1 | ER Diagram, Normalization, Schema Diagram |
| 2.Md. Nafiul Haque | 22-46355-1 | User Interface Planning, Table Creation Using SQL, Conclusion |
| 3.Mohammad Istishad Alam Tishad | 22-46130-1 | Data Insertion, Basic PL/SQL |
| 4.Antor Chandra Das | 21-45849-3 | Introduction, Project Proposal, Scenario Description, Advance PL/SQL |

Date of Submission: Aug 25, 2025

TABLE OF CONTENTS

| TOPICS | <i>Page no.</i> |
|-----------------------------|------------------------|
| I. Title Page | 1 |
| II. Table of Content | 2 |
| 1. Introduction | 3 |
| 2. Project Proposal | 3-4 |
| 3. User Interface Planning | 5-11 |
| 4. Scenario Description | 12 |
| 5. ER Diagram | 13 |
| 6. Normalization | 14-17 |
| 7. Schema Diagram | 18 |
| 8. Table Creation Using SQL | 19-25 |
| 9. Data Insertion Using SQL | 26-41 |
| 10. Basic PL/SQL | 42-56 |
| 11. Advance PL/SQL | 56-76 |
| 12. Conclusion | 77 |

Introduction

The International Trade Fair Committee (ITFC) convenes a series of large-scale trade exhibitions each year across multiple metropolitan centres. These events serve as significant platforms where corporate entities, emerging startups, and independent entrepreneurs converge as exhibitors, alongside an extensive influx of registered visitors. The administration of such fairs is inherently complex, as it necessitates the meticulous management of diverse information streams, including exhibitor credentials, stall allocations, visitor demographics, product catalogs, financial transactions, and post-event feedback. When handled manually or through fragmented systems, these processes are highly susceptible to inefficiencies, data inaccuracies, and impediments in generating comprehensive analytical reports. To address these challenges, this project proposes the development of a centralized database management framework designed to streamline, automate, and integrate the operational workflows of trade fairs conducted under the auspices of the ITFC.

Project Proposal

| Objectives | Scope | Methodology | Deliverables |
|--|---|---|---|
| ● To create a centralized database that ensures data integrity and eliminates redundancy across all operational areas. | Fair Management: Handling multiple trade fairs, each with specific names, cities, and dates. | 1. Scenario Analysis: Define the system's requirements and entities based on the operational flow of the ITFC. | A complete project report including an Introduction, Project Proposal, User Interface Planning, and Scenario Description. |
| ● To automate the process of exhibitor registration, stall allotment, and booking payment processing. | Venue Management: Managing halls and the assignment of stalls within them. Each stall is assigned to a single exhibitor per fair. | 2. Conceptual Design: Create an Entity-Relationship (ER) Diagram to visually represent the database entities and their relationships. | A detailed ER Diagram representing the conceptual data model. |
| ● To streamline visitor registration and the recording of ticket information. | Exhibitor Management: Storing exhibitor details, managing product information (name, category, price, stock), | 3. Logical Design (Normalization): Transform the conceptual model into a set of normalized | A full description of the Normalization process for all relations. |

| | | | |
|---|--|---|--|
| | and tracking payments (mode, transaction ID, status). | relations. The process will ensure that all tables are in at least Third Normal Form (3NF) to reduce data redundancy and improve data integrity. | |
| ● To effectively manage and track exhibitor products, including categories, descriptions, and stock levels. | Visitor Management: Capturing visitor registration data (name, contact, interests) and information about their tickets (type, cost, purchase date). | 4. Schema Design: Develop a final schema diagram illustrating the table structures and the foreign key relationships between them. | A final Schema Diagram of the database. |
| ● To capture and store visitor feedback and exhibitor sales data for future analysis and decision-making. | Post-Event Analytics: Recording visitor feedback, attendance figures, and exhibitor sales summaries to guide future planning. | 5. Database Implementation: Create the database tables using SQL based on the normalized schema. | SQL scripts for Table Creation and Data Insertion. |
| ● To enable the generation of complex reports, such as overall sales by category and busiest stalls, by implementing advanced PL/SQL features like stored procedures, triggers, and packages. | | 6. Data Population: Insert sample data into the tables using SQL to test the database structure and relationships. | |
| | | 7. Advanced Functionality: Implement sophisticated queries and business logic using PL/SQL features like stored procedures, triggers, and packages to support reporting and data management tasks. | |

User Interface Planning

Welcome to ITFC Trade Fair Platform

Comprehensive platform for managing international trade fairs, connecting exhibitors with visitors, and streamlining event operations.

ITFC Staff Dashboard
Complete control over trade fair operations, exhibitor management, and analytics

- Fair & Hall Management
- Exhibitor Approval Process
- Revenue & Analytics Tracking
- Visitor Management

Exhibitor Portal
Comprehensive tools for exhibitors to manage their trade fair presence

- Product Catalog Management
- Stall Booking & Payments
- Sales Performance Tracking
- Lead Generation Tools

Visitor Experience
Seamless registration and engagement platform for trade fair attendees

- Easy Registration Process
- Digital Ticket Management
- Exhibitor Discovery
- Feedback & Rating System

Sign In to Your Account

Access your trade fair dashboard

[Sign In](#) [Sign Up](#)

Email Address: Enter your email

Password: Enter your password

Remember me [Forgot password?](#)

[Sign In](#)

Try Demo Accounts:

| | |
|------------|---------------------------------|
| Admin: | admin@itfc.com / admin123 |
| Exhibitor: | exhibitor@company.com / expo123 |
| Visitor: | visitor@email.com / visit123 |

Administrator Dashboard

Manage trade fairs, exhibitors, and track performance

Active Fairs: 3 (+2 from last month)

Total Exhibitors: 156 (23 pending approval, +12%)

Registered Visitors: 2,847 (+24%)

Total Revenue: \$485,320 (From stall bookings, +18%)

[Overview](#) [Fair Management](#) [Exhibitors](#) [Halls & Stalls](#) [Analytics](#)

Recent Activities
Latest updates and actions

- Exhibitor "Tech Solutions Inc" approved 2 hours ago
- New fair "Global Fashion Week" created 5 hours ago
- Payment received for Stall #A12 1 day ago

Pending Approvals
Exhibitors waiting for approval

| | |
|---|--|
| Innovation Labs Ltd Submitted 2 days ago | View Approve |
| Green Energy Corp Submitted 3 days ago | View Approve |

Occupancy Rate 87% 

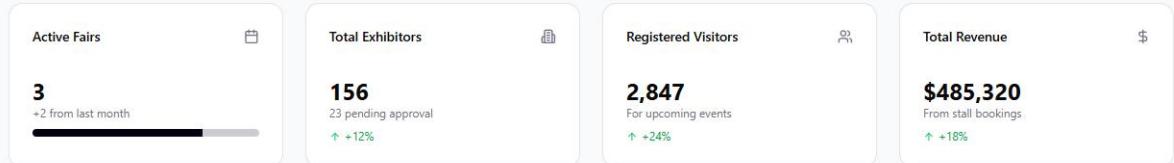
Avg Rating 4.6 

Growth Rate +23% 

77

Administrator Dashboard

Manage trade fairs, exhibitors, and track performance



Overview

Fair Management

Exhibitors

Halls & Stalls

Analytics

Exhibitor Management

Export List

Send Notifications

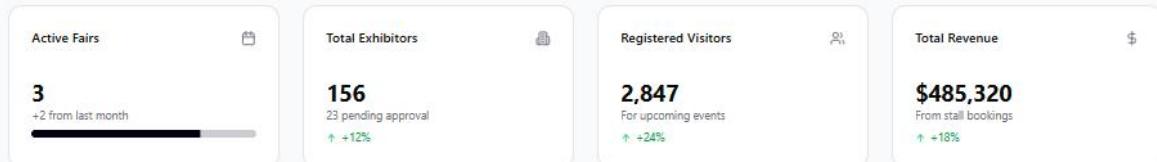
Exhibitor Applications

Review and manage exhibitor registrations

| Company | Contact Person | Status | Applied Date | Actions |
|--|---|----------|--------------|-----------------|
| Tech Solutions Inc AI & Machine Learning | John Smith john@techsolutions.com | Approved | Mar 1, 2024 | View Contact |
| Innovation Labs Ltd Robotics & Automation | Sarah Johnson sarah@innovationlabs.com | Pending | Mar 3, 2024 | Approve Reject |

Administrator Dashboard

Manage trade fairs, exhibitors, and track performance



Overview

Fair Management

Exhibitors

Halls & Stalls

Analytics

Trade Fair Management

Create New Fair

All Trade Fairs

Manage and monitor all trade fair events

Search fairs...

Filter

Export

International Tech Expo 2024

New York Convention Center • March 15-17, 2024

The premier technology exhibition featuring the latest innovations in AI, IoT, and digital transformation.

Exhibitors: 89/120

Revenue: \$234,500

Visitors: 1,247



Global Fashion Week

Paris Expo • April 20-25, 2024

Showcasing the latest trends in fashion, textiles, and lifestyle products from around the world.

Exhibitors: 45/200

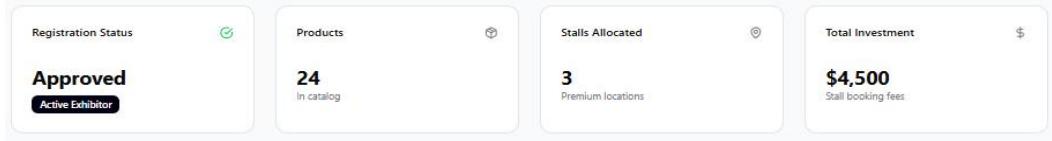
Revenue: \$67,800

Visitors: 342



Exhibitor Dashboard

Manage your products, stalls, and track performance



Overview

Products

My Stalls

Payments

Performance

Product Catalog

+ Add New Product

Showcase Products

Manage products for exhibition

Search products...

Filter

Add New Product

Click to add

AI Vision Camera

Featured

Advanced AI-powered surveillance camera with real-time analysis

\$599.99

IoT Gateway Hub

New

Smart gateway for connecting industrial IoT devices

\$299.99

Exhibitor Dashboard

Manage your products, stalls, and track performance

Registration Status

Approved

(Active Exhibitor)

Products

24

In catalog

Stalls Allocated

3

Premium locations

Total Investment

\$4,500

Stall booking fees

Overview

Products

My Stalls

Payments

Performance

Company Information

Your registered business details

Company Name

Tech Solutions Inc

Contact Person

John Smith

Email

john@techsolutions.com

Phone

+1-555-0123

Business Category

AI & Machine Learning Solutions

Current Participation

Active trade fair events

International Tech Expo 2024

Location: New York

Dates: Mar 15-17

Hall: Tech Pavilion A

Stalls: A12, A13, A14

Allocated Stalls

Your exhibition spaces



Stall #A12

Hall A - Corner Position

Premium



Stall #A13

Hall A - Center Row

Standard



Stall #A14

Hall A - Center Row

Standard

Visitor Dashboard

Explore trade fairs and manage your experience

Registration Status



Active

Verified Visitor

Current Ticket



Premium

Full access pass

Feedback Given



3

Reviews submitted

My Profile

My Tickets

Trade Fairs

Feedback

Personal Information

Your registration details

Full Name

Sarah Johnson

Email Address

sarah.johnson@email.com

Phone Number

+1-555-0456

Areas of Interest

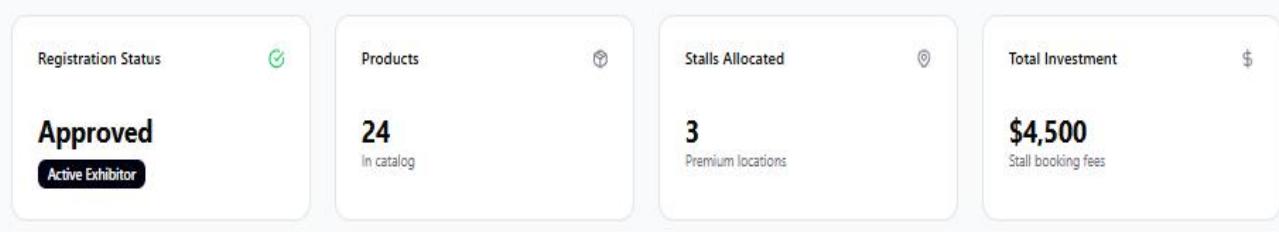
Technology, Innovation, Sustainable Development, Digital Marketing

Member Since

February 2024

Exhibitor Dashboard

Manage your products, stalls, and track performance



Overview

Products

My Stalls

Payments

Performance

Payment Management

 Make Payment

Payment Summary

Your booking and payment overview

Total Investment **\$4,500**

Stall A12 (Premium): \$1,500

Stall A13 (Standard): \$1,200

Stall A14 (Standard): \$1,200

Additional Services: \$600

Paid

\$4,500

Pending

\$0

Saved Payment Methods

Manage your payment options



Expires 12/2026

Primary

+ Add Payment Method

Payment History

Track all your transactions

| Transaction ID | Description | Amount | Status | Date |
|----------------|-------------------------------|------------|-----------|-------------|
| TXN_1234567890 | Stall A12 - Premium Location | \$1,500.00 | Completed | Mar 1, 2024 |
| TXN_1234567891 | Stall A13 - Standard Location | \$1,200.00 | Completed | Mar 2, 2024 |

Visitor Dashboard

Explore trade fairs and manage your experience

Registration Status

Active

Verified Visitor



Current Ticket

Premium

Full access pass



Feedback Given

3

Reviews submitted



My Profile

My Tickets

Trade Fairs

Feedback

Available Trade Fairs

International Tech Expo 2024 Ongoing

📍 New York Convention Center • March 15-17, 2024

Discover the latest innovations in artificial intelligence, IoT, robotics, and digital transformation. Connect with industry leaders and explore cutting-edge technologies.

89
Exhibitors

1,200+
Visitors

4.6
Avg Rating

View Floor Plan

Event Schedule

Exhibitor List

Browse All

Global Fashion Week Upcoming

📍 Paris Exhibition Center • April 20-25, 2024

Explore the world of fashion, textiles, and lifestyle products. Meet designers, discover trends, and network with fashion industry professionals from around the globe.

145
Exhibitors

800+
Expected

-
Rating

Register Interest

Add to Calendar

Visitor Dashboard

Explore trade fairs and manage your experience

Registration Status



Active

Verified Visitor

Current Ticket



Premium

Full access pass

Feedback Given

3

Reviews submitted



My Profile

My Tickets

Trade Fairs

Feedback

Digital Tickets

Your trade fair access passes

Premium Access Pass

ITFC Official Visitor Ticket

Visitor Name
Sarah Johnson

Ticket ID
VIS-2024-7891

Valid For
All Current Events

Access Level
Premium



Premium Benefits

- Priority access to all halls
- VIP networking events
- Complimentary refreshments
- Free digital catalog
- Product demonstration access

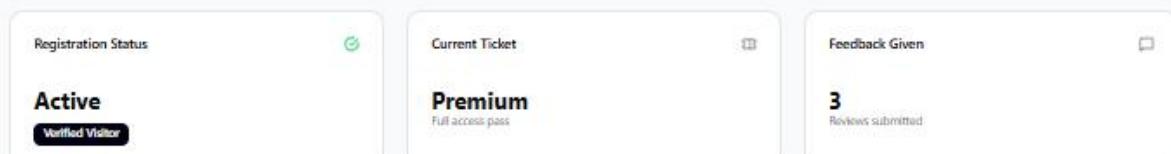
Event Schedule

- Tech Expo Opening: Mar 15, 9:00 AM
Fashion Week: Apr 20, 10:00 AM
Networking Event: Mar 16, 6:00 PM

of 77

Visitor Dashboard

Explore trade fairs and manage your experience



[My Profile](#)

[My Tickets](#)

[Trade Fairs](#)

[Feedback](#)

Feedback & Reviews

[Submit New Feedback](#)

Share Your Experience

Help us improve future trade fairs

Select Trade Fair

Choose an event to review

Overall Rating

(4/5)

Your Comments

Share your thoughts about the event, organization, exhibitors, etc.

[Submit Feedback](#)

Your Previous Reviews

Feedback you've submitted

International Tech Expo 2024

Mar 17, 2024

(5/5)

Excellent organization and fascinating displays of cutting-edge technology. The networking opportunities were outstanding, and I learned about many innovative solutions.

Previous Fashion Expo 2023

Nov 15, 2023

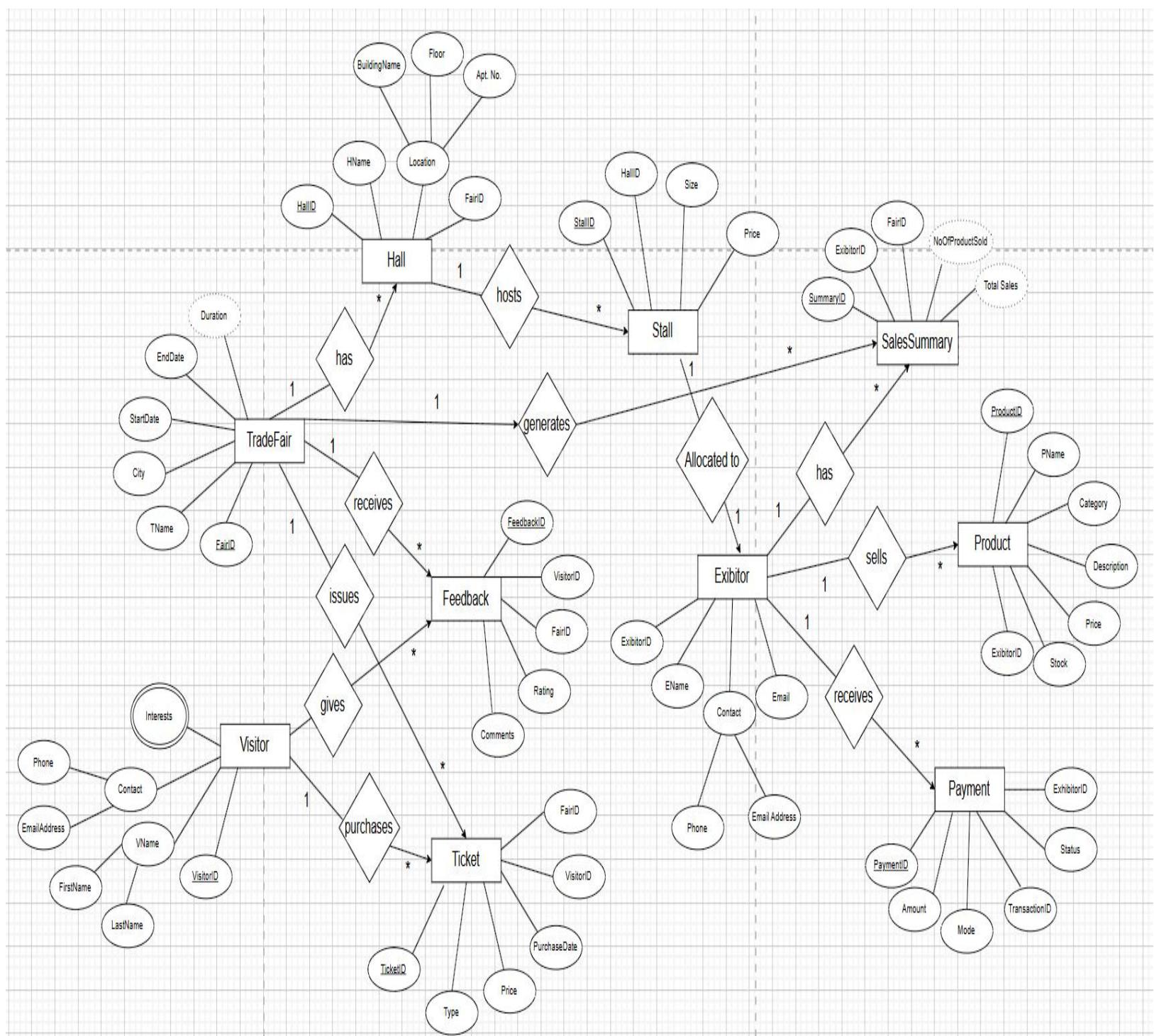
(4/5)

Great variety of exhibitors and beautiful displays. Could have been better organized in terms of crowd management, but overall a very inspiring event.

Scenario Description

In a Trade-Fair Management System, a trade fair is identified by a FairID. It has many halls and one hall belongs to exactly one trade fair. The system also stores trade-fair name, city, start date, end date and duration. A hall hosts many stalls and one stall is hosted by exactly one hall. A hall is identified by a HallID, its location is composed of building name, floor and apartment number, and the system stores hall name. A stall is allocated to exactly one exhibitor and an exhibitor may be allocated many stalls. A stall is identified by a StallID and the system stores it's stall size and price. An exhibitor sells many products and a product is sold by exactly one exhibitor. An exhibitor is identified by an ExhibitorID and the system also stores exhibitor name, contact, phone and email address. A product is identified by a ProductID and the system stores product name, category, description, price and stock. An exhibitor receives many payments and a payment is made for exactly one exhibitor. A payment is identified by a PaymentID and the system stores transaction ID, amount, mode and status. A trade fair generates many sales summaries where each sales summary belongs to exactly one trade fair and one exhibitor. A sales summary is identified by a SummaryID and the system stores number of products sold and total sales. A visitor may purchase many tickets and a ticket is purchased by exactly one visitor. A visitor is identified by a VisitorID and the system stores visitor name, contact, phone, email address, interests, first name and last name. A ticket is identified by a TicketID and the system stores ticket type, price and purchase date. A visitor can give one or many feedbacks and a trade fair may receive many feedbacks. A feedback belongs to exactly one visitor and one trade fair. A feedback is identified by a FeedbackID and the system stores rating and comments.

ER Diagram



Normalization

Relation1: HAS (TradeFair has Hall)

UNF:

TradeFair(FairID, FName, City, StartDate, EndDate, Duration, HallID, HName, Location, BuildingName, Floor, AptNo)

1NF:

No multivalued attributes in 1NF.

FairID, FName, City, StartDate, EndDate, Duration, HallID, HName, Location, BuildingName, Floor, AptNo

2NF:

Remove partial dependencies (Hall depends only on HallID).

1. FairID, Name, City, StartDate, EndDate, Duration

2. HallID, HName, Location, BuildingName, Floor, AptNo, FairID

3NF:

No transitive dependency.

1. FairID, FName, City, StartDate, EndDate, Duration

2. HallID, HName, Location, BuildingName, Floor, AptNo, FairID

Table Creation:

TradeFair(FairID, Name, City, StartDate, EndDate, Duration)

Hall(HallID, HName, Location, BuildingName, Floor, AptNo, FairID)

Relation2: HOSTS (Hall hosts Stall)

UNF:

Hall(HallID, HName, Location, BuildingName, Floor, AptNo, StallID, Size, Price)

1NF:

Atomic attributes only.

HallID, HName, Location, BuildingName, Floor, AptNo, StallID, Size, Price

2NF:

Remove partial dependencies (Stall depends only on StallID).

1. HallID, HName, Location, BuildingName, Floor, AptNo

2. StallID, Size, Price, HallID

3NF:

No transitive dependency.

1. HallID, HName, Location, BuildingName, Floor, AptNo

2. StallID, Size, Price, HallID

Table Creation:

Hall(HallID, HName, Location, BuildingName, Floor, AptNo)

Stall(StallID, Size, Price, HallID)

Relation3: GENERATES (TradeFair generates SalesSummary)

UNF:

TradeFair(FairID, FName, City, StartDate, EndDate, Duration, SummaryID, ExhibitorID, NoOfProductsSold, TotalSales)

1NF:

Already atomic.

FairID, FName, City, StartDate, EndDate, Duration, SummaryID, ExhibitorID, NoOfProductsSold, TotalSales

2NF:

Removes partial dependency.

1. FairID, FName, City, StartDate, EndDate, Duration

2. SummaryID, FairID, ExhibitorID, NoOfProductsSold, TotalSales

3NF:

No transitive dependency.

1. FairID, FName, City, StartDate, EndDate, Duration

2. SummaryID, FairID, ExhibitorID, NoOfProductsSold, TotalSales

Table Creation:

TradeFair(FairID, FName, City, StartDate, EndDate, Duration)

SalesSummary(SummaryID, FairID, ExhibitorID, NoOfProductsSold, TotalSales)

Relation4: HAS (SalesSummary has Exhibitor)

UNF:

SalesSummary(SummaryID, FairID, ExhibitorID, EName, Contact, Email, Phone, EmailAddress)

1NF:

Already atomic.

SummaryID, FairID, ExhibitorID, EName, Contact, Email, Phone, EmailAddress

2NF:

Exhibitor info depends only on ExhibitorID.

1. SummaryID, FairID, ExhibitorID

2. ExhibitorID, EName, Contact, Email, Phone, EmailAddress

3NF:

No transitive dependency.

1. SummaryID, FairID, ExhibitorID

2. ExhibitorID, EName, Contact, Email, Phone, EmailAddress

Table Creation:

SalesSummary(SummaryID, FairID, ExhibitorID)

Exhibitor(ExhibitorID, EName, Contact, Email, Phone, EmailAddress)

Relation5: SELLS (Exibitor sells Product)

UNF:

Exhibitor(ExhibitorID, EName, Contact, Email, Phone, EmailAddress, ProductID, PName, Category, Description, Price, Stock)

1NF:

Already Atromic.

ExhibitorID, EName, Contact, Email, Phone, EmailAddress, ProductID, PName, Category, Description, Price, Stock

2NF:

Product depends only on ProductID.

1. ExhibitorID, EName, Contact, Email, Phone, EmailAddress

2. ProductID, PName, Category, Description, Price, Stock, ExhibitorID

3NF:

No transitive dependency.

1. ExhibitorID, EName, Contact, Email, Phone, EmailAddress
 2. ProductID, PName, Category, Description, Price, Stock, ExhibitorID
- Table Creation:
- Exhibitor(ExhibitorID, EName, Contact, Email, Phone, EmailAddress)
Product(ProductID, PName, Category, Description, Price, Stock, ExhibitorID)

Relation6: RECEIVES (Exibitor receives Payment)

UNF:

Exhibitor(ExhibitorID, EName, Contact, Email, Phone, EmailAddress, PaymentID, Amount, Mode, TransactionID, Status)

1NF:

Already atomic.

ExhibitorID, Name, Contact, Email, Phone, EmailAddress, PaymentID, Amount, Mode, TransactionID, Status

2NF:

Payment depends only on PaymentID.

1. ExhibitorID, EName, Contact, Email, Phone, EmailAddress
2. PaymentID, Amount, Mode, TransactionID, Status, ExhibitorID

3NF:

1. ExhibitorID, EName, Contact, Email, Phone, EmailAddress
2. PaymentID, Amount, Mode, TransactionID, Status, ExhibitorID

Table Creation:

Exhibitor(ExhibitorID, EName, Contact, Email, Phone, EmailAddress)

Payment(PaymentID, Amount, Mode, TransactionID, Status, ExhibitorID)

Relation7: PURCHASES (Visitor purchases Tickets)

UNF:

Visitor(VisitorID, FirstName, LastName, Contact, Phone, EmailAddress, Interests, TicketID, Type, Price, PurchaseDate, FairID)

1NF:

Interests is a multi-valued attribute.

1. Visitor(VisitorID, FirstName, LastName, Contact, Phone, EmailAddress)
2. VisitorInterest(VisitorID, Interest)
3. Ticket(TicketID, Type, Price, PurchaseDate, FairID, VisitorID)

2NF:

No partial dependency remains.

1. Visitor(VisitorID, FirstName, LastName, Contact, Phone, EmailAddress)
2. VisitorInterest(VisitorID, Interest)
3. Ticket(TicketID, Type, Price, PurchaseDate, FairID, VisitorID)

3NF:

No Transitive dependency.

1. Visitor(VisitorID, FirstName, LastName, Contact, Phone, EmailAddress)
2. VisitorInterest(VisitorID, Interest)
3. Ticket(TicketID, Type, Price, PurchaseDate, FairID, VisitorID)

Table Creation:

1. Visitor(VisitorID, FirstName, LastName, Contact, Phone, EmailAddress)
2. VisitorInterest(VisitorID, Interest)
3. Ticket(TicketID, Type, Price, PurchaseDate, FairID, VisitorID)

Relation8: GIVES (Visitor gives Feedback)

UNF:

Visitor(VisitorID, FirstName, LastName, Contact, Phone, EmailAddress, Interests, FeedbackID, FairID, Rating, Comments)

1NF:

Interests is a multi-valued attribute.

1. Visitor(VisitorID, FirstName, LastName, Contact, Phone, EmailAddress)
2. VisitorInterest(VisitorID, Interest)
3. Feedback(FeedbackID, FairID, VisitorID, Rating, Comments)

2NF:

No partial dependency remains.

1. Visitor(VisitorID, FirstName, LastName, Contact, Phone, EmailAddress)
2. VisitorInterest(VisitorID, Interest)
3. Feedback(FeedbackID, FairID, VisitorID, Rating, Comments)

3NF:

No Transitive dependency.

1. Visitor(VisitorID, FirstName, LastName, Contact, Phone, EmailAddress)
2. VisitorInterest(VisitorID, Interest)
3. Feedback(FeedbackID, FairID, VisitorID, Rating, Comments)

Table Creation:

Visitor(VisitorID, FirstName, LastName, Contact, Phone, EmailAddress)

VisitorInterest(VisitorID, Interest)

Feedback(FeedbackID, FairID, VisitorID, Rating, Comments)

Final Tables:

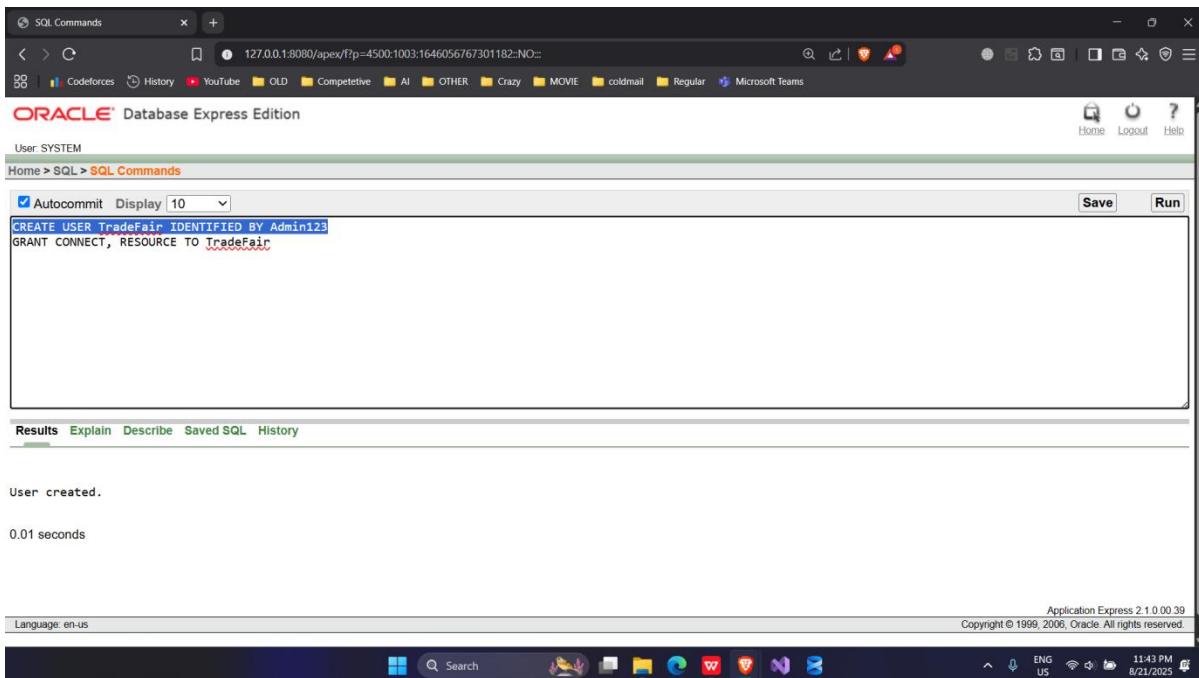
1. TradeFair(FairID, TName, City, StartDate, EndDate, Duration)
2. Hall(HallID, HName, Location, BuildingName, Floor, AptNo, FairID)
3. Stall(StallID, Size, Price, HallID)
4. SalesSummary(SummaryID, FairID, ExhibitorID, NoOfProductsSold, TotalSales)
5. Exhibitor(ExhibitorID, EName, Contact, Email, Phone, EmailAddress)
6. Product(ProductID, PName, Category, Description, Price, Stock, ExhibitorID)
7. Payment(PaymentID, Amount, Mode, TransactionID, Status, ExhibitorID)
8. Visitor(VisitorID, FirstName, LastName, Contact, Phone, EmailAddress, Interests)
9. VisitorInterest(VisitorID, Interest)
10. Ticket(TicketID, Type, Price, PurchaseDate, FairID, VisitorID)
11. Feedback(FeedbackID, FairID, VisitorID, Rating, Comments)

Schema Diagram



Table Creation Using SQL

- CREATE USER TradeFair IDENTIFIED BY Admin123

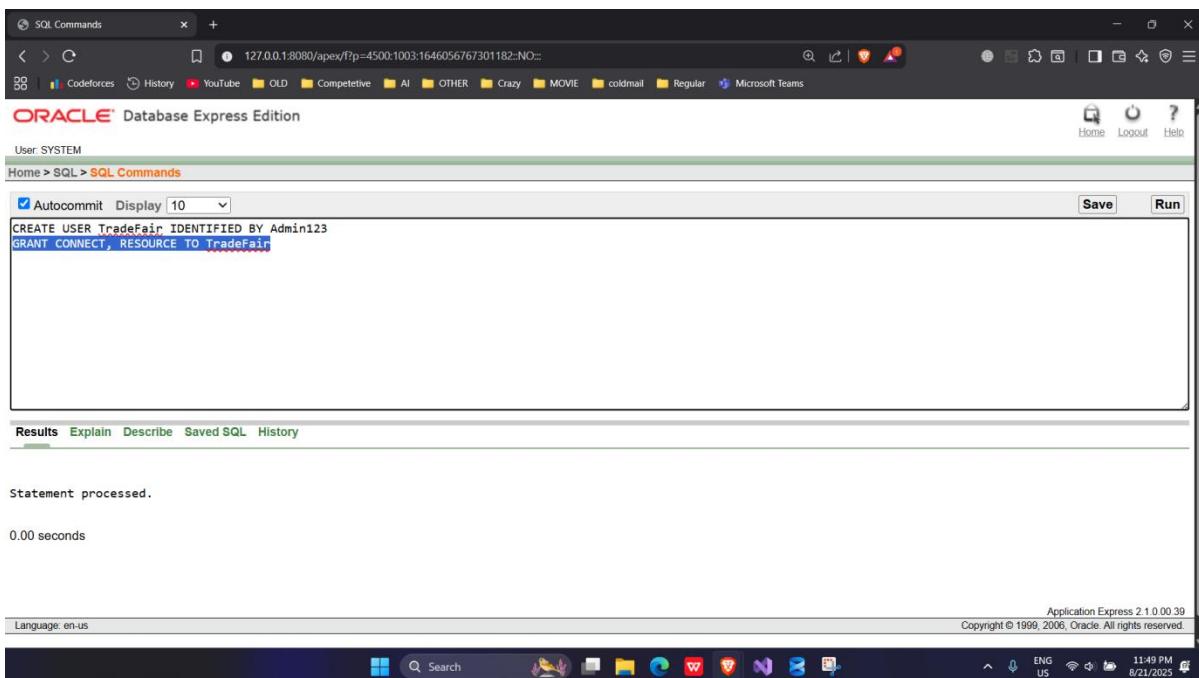


The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
CREATE USER TradeFair IDENTIFIED BY Admin123
GRANT CONNECT, RESOURCE TO TradeFair;
```

The results pane shows the output: "User created." and "0.01 seconds". The status bar at the bottom right indicates "Application Express 2.1.0.00.39" and the date "8/21/2025".

- GRANT CONNECT, RESOURCE TO TradeFair



The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
CREATE USER TradeFair IDENTIFIED BY Admin123
GRANT CONNECT, RESOURCE TO TradeFair;
```

The results pane shows the output: "Statement processed." and "0.00 seconds". The status bar at the bottom right indicates "Application Express 2.1.0.00.39" and the date "8/21/2025".

-- 1. TradeFair

CREATE TABLE TradeFair (

```
FairID      NUMBER PRIMARY KEY,  
TName       VARCHAR2(100),  
City        VARCHAR2(50),  
StartDate   DATE,  
EndDate     DATE,  
Duration    NUMBER  
);
```

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
CREATE TABLE TradeFair (  
    FairID      NUMBER PRIMARY KEY,  
    TName       VARCHAR2(100),  
    City        VARCHAR2(50),  
    StartDate   DATE,  
    EndDate     DATE,  
    Duration    NUMBER  
)
```

Below the code, the results show:

```
Table created.  
0.02 seconds
```

At the bottom, the status bar indicates "Application Express 2.1.0.0.39" and "Copyright © 1999, 2006, Oracle. All rights reserved."

-- 2. Hall

CREATE TABLE Hall (

```
HallID      NUMBER PRIMARY KEY,  
HName       VARCHAR2(100),  
Location    VARCHAR2(100),  
BuildingName VARCHAR2(100),  
Floor       NUMBER,  
AptNo      VARCHAR2(20),  
FairID      NUMBER,  
FOREIGN KEY (FairID) REFERENCES TradeFair(FairID)  
);
```

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
CREATE TABLE Hall (  
    HallID      NUMBER PRIMARY KEY,  
    HName       VARCHAR2(100),  
    Location    VARCHAR2(100),  
    BuildingName VARCHAR2(100),  
    Floor       NUMBER,  
    AptNo      VARCHAR2(20),  
    FairID      NUMBER,  
    FOREIGN KEY (FairID) REFERENCES TradeFair(FairID)  
)
```

Below the code, the results show:

```
Table created.  
0.00 seconds
```

At the bottom, the status bar indicates "Application Express 2.1.0.0.39" and "Copyright © 1999, 2006, Oracle. All rights reserved."

-- 3. Stall

CREATE TABLE Stall (

StallID NUMBER PRIMARY KEY,

S_size VARCHAR2(50),

Price NUMBER(10,2),

HallID NUMBER,

FOREIGN KEY (HallID) REFERENCES Hall(HallID)

);

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
CREATE TABLE Stall (
    StallID NUMBER PRIMARY KEY,
    S_size VARCHAR2(50),
    Price NUMBER(10,2),
    HallID NUMBER,
    FOREIGN KEY (HallID) REFERENCES Hall(HallID)
);
```

The results pane shows the message "Table created." and "0.00 seconds". The bottom status bar indicates "Application Express 2.1.0.00.39" and the date "8/22/2025".

-- 4. Exhibitor

CREATE TABLE Exhibitor (

ExhibitorID NUMBER PRIMARY KEY,

EName VARCHAR2(100),

Contact VARCHAR2(100),

Email VARCHAR2(100),

Phone VARCHAR2(20),

EmailAddress VARCHAR2(100)

);

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
-- 5. Exhibitor
CREATE TABLE Exhibitor (
    ExhibitorID NUMBER PRIMARY KEY,
    EName VARCHAR2(100),
    Contact VARCHAR2(100),
    Email VARCHAR2(100),
    Phone VARCHAR2(20),
    EmailAddress VARCHAR2(100)
);
```

The results pane shows the message "Table created." and "0.00 seconds". The bottom status bar indicates "Application Express 2.1.0.00.39" and the date "8/22/2025".

-- 5. SalesSummary

```
CREATE TABLE SalesSummary (
```

```
    SummaryID      NUMBER PRIMARY KEY,
```

```
    FairID        NUMBER,
```

```
    ExhibitorID   NUMBER,
```

```
    NoOfProductsSold NUMBER,
```

```
    TotalSales     NUMBER(12,2),
```

```
    FOREIGN KEY (FairID) REFERENCES TradeFair(FairID),
```

```
    FOREIGN KEY (ExhibitorID) REFERENCES Exhibitor(ExhibitorID)
```

```
);
```

-- 6. Product

```
CREATE TABLE Product (
```

```
    ProductID     NUMBER PRIMARY KEY,
```

```
    PName         VARCHAR2(100),
```

```
    Category       VARCHAR2(50),
```

```
    Description    VARCHAR2(200),
```

```
    Price          NUMBER(10,2),
```

```
    Stock           NUMBER,
```

```
    ExhibitorID   NUMBER,
```

```
    FOREIGN KEY (ExhibitorID) REFERENCES Exhibitor(ExhibitorID)
```

```
);
```

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
CREATE TABLE SalesSummary (
    SummaryID      NUMBER PRIMARY KEY,
    FairID        NUMBER,
    ExhibitorID   NUMBER,
    NoOfProductsSold NUMBER,
    TotalSales     NUMBER(12,2),
    FOREIGN KEY (FairID) REFERENCES TradeFair(FairID),
    FOREIGN KEY (ExhibitorID) REFERENCES Exhibitor(ExhibitorID)
);
```

The results pane shows the message "Table created." and a duration of "0.01 seconds". The status bar at the bottom right indicates "Application Express 2.1.0.0.39", "Copyright © 1999, 2006, Oracle. All rights reserved.", and the date "8/22/2025".

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
CREATE TABLE Product (
    ProductID     NUMBER PRIMARY KEY,
    PName         VARCHAR2(100),
    Category       VARCHAR2(50),
    Description    VARCHAR2(200),
    Price          NUMBER(10,2),
    Stock           NUMBER,
    ExhibitorID   NUMBER,
    FOREIGN KEY (ExhibitorID) REFERENCES Exhibitor(ExhibitorID)
);
```

The results pane shows the message "Table created." and a duration of "0.01 seconds". The status bar at the bottom right indicates "Application Express 2.1.0.0.39", "Copyright © 1999, 2006, Oracle. All rights reserved.", and the date "8/22/2025".

-- 7. Payment

```
CREATE TABLE Payment (
    PaymentID NUMBER PRIMARY KEY,
    Amount NUMBER(12,2),
    PaymentMode VARCHAR2(50),
    TransactionID VARCHAR2(100),
    Status VARCHAR2(50),
    ExhibitorID NUMBER,
    FOREIGN KEY (ExhibitorID) REFERENCES Exhibitor(ExhibitorID)
);
```

The screenshot shows the Oracle Database Express Edition interface. In the top navigation bar, it says "User TRADEFAIR". Below that, the path "Home > SQL > SQL Commands" is visible. The main area contains the SQL code for creating the Payment table. At the bottom of the code editor, there are tabs for "Results", "Explain", "Describe", "Saved SQL", and "History". The status bar at the bottom right shows "Application Express 2.1.0.00.39", "Language: en-us", "Copyright © 1999, 2006, Oracle. All rights reserved.", and a timestamp of "8/22/2025 8:27 AM".

-- 8. Visitor

```
CREATE TABLE Visitor (
    VisitorID NUMBER PRIMARY KEY,
    FirstName VARCHAR2(50),
    LastName VARCHAR2(50),
    Contact VARCHAR2(100),
    Phone VARCHAR2(20),
    EmailAddress VARCHAR2(100),
    Interests VARCHAR2(200)
);
```

The screenshot shows the Oracle Database Express Edition interface. In the top navigation bar, it says "User TRADEFAIR". Below that, the path "Home > SQL > SQL Commands" is visible. The main area contains the SQL code for creating the Visitor table. At the bottom of the code editor, there are tabs for "Results", "Explain", "Describe", "Saved SQL", and "History". The status bar at the bottom right shows "Application Express 2.1.0.00.39", "Language: en-us", "Copyright © 1999, 2006, Oracle. All rights reserved.", and a timestamp of "8/22/2025 8:31 PM".

SQL Commands

User: TRADEFAIR

Home > SQL > SQL Commands

Autocommit Display 10 Save Run

```
CREATE TABLE VisitorInterest (
    VisitorID NUMBER,
    Interest VARCHAR2(100),
    PRIMARY KEY (VisitorID, Interest),
    FOREIGN KEY (VisitorID) REFERENCES Visitor(VisitorID)
);
```

Results Explain Describe Saved SQL History

Table created.

0.01 seconds

Language: en-us Copyright © 1999, 2006, Oracle. All rights reserved.

Application Express 2.1.0.0.39
8:32 PM 8/22/2025

-- 9. VisitorInterest

CREATE TABLE VisitorInterest (

 VisitorID NUMBER,

 Interest VARCHAR2(100),

 PRIMARY KEY (VisitorID, Interest),

 FOREIGN KEY (VisitorID) REFERENCES Visitor(VisitorID)

);

-- 10. Ticket

CREATE TABLE Ticket (

 TicketID NUMBER PRIMARY KEY,

 Type VARCHAR2(50),

 Price NUMBER(10,2),

 PurchaseDate DATE,

 FairID NUMBER,

 VisitorID NUMBER,

 FOREIGN KEY (FairID) REFERENCES TradeFair(FairID),

 FOREIGN KEY (VisitorID) REFERENCES Visitor(VisitorID)

);

SQL Commands

User: TRADEFAIR

Home > SQL > SQL Commands

Autocommit Display 10 Save Run

```
CREATE TABLE Ticket (
    TicketID NUMBER PRIMARY KEY,
    Type VARCHAR2(50),
    Price NUMBER(10,2),
    PurchaseDate DATE,
    FairID NUMBER,
    VisitorID NUMBER,
    FOREIGN KEY (FairID) REFERENCES TradeFair(FairID),
    FOREIGN KEY (VisitorID) REFERENCES Visitor(VisitorID)
);
```

Results Explain Describe Saved SQL History

Table created.

0.02 seconds

Language: en-us Copyright © 1999, 2006, Oracle. All rights reserved.

Application Express 2.1.0.0.39
8:34 PM 8/22/2025

-- 11. Feedback

CREATE TABLE Feedback (

FeedbackID NUMBER PRIMARY KEY,

FairID NUMBER,

VisitorID NUMBER,

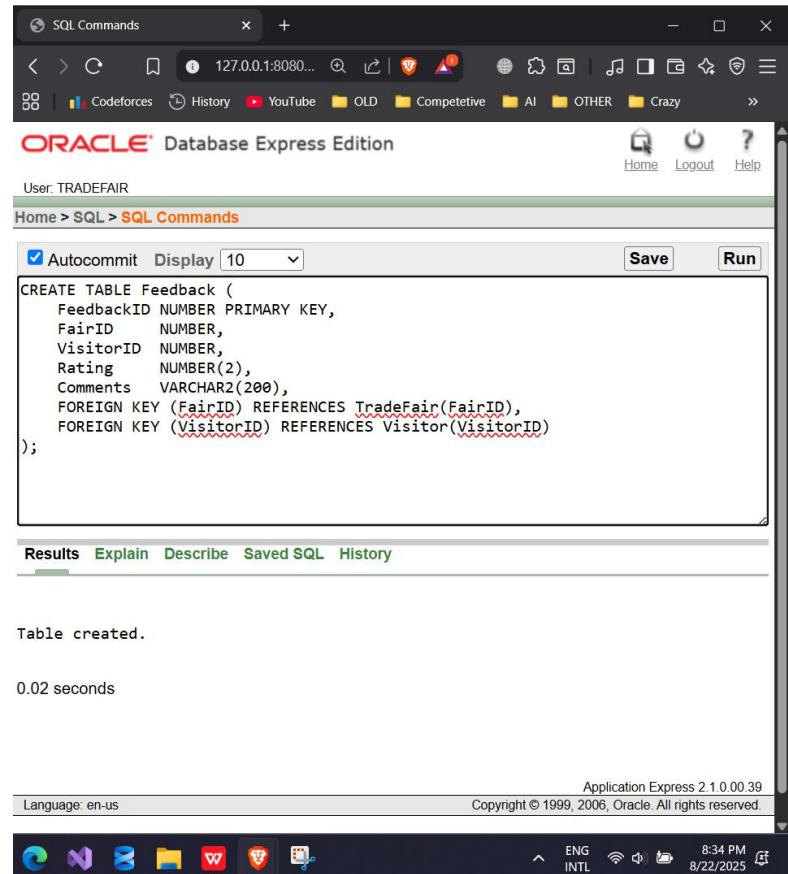
Rating NUMBER(2),

Comments VARCHAR2(200),

FOREIGN KEY (FairID) REFERENCES TradeFair(FairID),

FOREIGN KEY (VisitorID) REFERENCES Visitor(VisitorID)

);



The screenshot shows the Oracle Database Express Edition SQL Commands interface. The SQL editor contains the following code:

```
CREATE TABLE Feedback (
    FeedbackID NUMBER PRIMARY KEY,
    FairID      NUMBER,
    VisitorID   NUMBER,
    Rating      NUMBER(2),
    Comments    VARCHAR2(200),
    FOREIGN KEY (FairID) REFERENCES TradeFair(FairID),
    FOREIGN KEY (VisitorID) REFERENCES Visitor(VisitorID)
);
```

The results pane shows the message "Table created." and a execution time of "0.02 seconds". The status bar at the bottom right indicates "Application Express 2.1.0.00.39", "Copyright © 1999, 2006, Oracle. All rights reserved", "Language: en-us", "ENG INTL", "8:34 PM 8/22/2025", and "8:34 PM 8/22/2025".

Data Insertion Using SQL

1) TradeFair table insertion :

```
INSERT INTO TradeFair VALUES (1, 'Tech Expo 2025', 'Dhaka', DATE '2025-01-10', DATE '2025-01-20', 10);
INSERT INTO TradeFair VALUES (2, 'Food Carnival', 'Chattogram', DATE '2025-02-05', DATE '2025-02-12', 7);
INSERT INTO TradeFair VALUES (3, 'Book Fair', 'Dhaka', DATE '2025-03-01', DATE '2025-03-15', 14);
INSERT INTO TradeFair VALUES (4, 'Startup Fair', 'Sylhet', DATE '2025-04-10', DATE '2025-04-15', 5);
INSERT INTO TradeFair VALUES (5, 'Furniture Expo', 'Rajshahi', DATE '2025-05-05', DATE '2025-05-10', 5);
INSERT INTO TradeFair VALUES (6, 'Car Show', 'Khulna', DATE '2025-06-01', DATE '2025-06-07', 6);
INSERT INTO TradeFair VALUES (7, 'Medical Tech Fair', 'Dhaka', DATE '2025-07-01', DATE '2025-07-10', 9);
INSERT INTO TradeFair VALUES (8, 'Education Expo', 'Chattogram', DATE '2025-08-15', DATE '2025-08-20', 5);
INSERT INTO TradeFair VALUES (9, 'Clothing Fair', 'Dhaka', DATE '2025-09-01', DATE '2025-09-05', 4);
INSERT INTO TradeFair VALUES (10, 'Sports Gear Fair', 'Sylhet', DATE '2025-10-01', DATE '2025-10-08', 7);
INSERT INTO TradeFair VALUES (11, 'Toy Expo', 'Rajshahi', DATE '2025-11-01', DATE '2025-11-05', 4);
INSERT INTO TradeFair VALUES (12, 'Mobile Fair', 'Dhaka', DATE '2025-12-01', DATE '2025-12-07', 6);
INSERT INTO TradeFair VALUES (13, 'IT Summit', 'Chattogram', DATE '2025-01-15', DATE '2025-01-20', 5);
INSERT INTO TradeFair VALUES (14, 'Agriculture Fair', 'Khulna', DATE '2025-02-20', DATE '2025-02-25', 5);
INSERT INTO TradeFair VALUES (15, 'Tourism Expo', 'CoxsBazar', DATE '2025-03-10', DATE '2025-03-14', 4);
INSERT INTO TradeFair VALUES (16, 'Energy Expo', 'Dhaka', DATE '2025-04-01', DATE '2025-04-06', 5);
INSERT INTO TradeFair VALUES (17, 'AI Expo', 'Dhaka', DATE '2025-05-15', DATE '2025-05-20', 5);
INSERT INTO TradeFair VALUES (18, 'Art Fair', 'Sylhet', DATE '2025-06-10', DATE '2025-06-15', 5);
INSERT INTO TradeFair VALUES (19, 'Gaming Expo', 'Dhaka', DATE '2025-07-15', DATE '2025-07-20', 5);
INSERT INTO TradeFair VALUES (20, 'Film Expo', 'Chattogram', DATE '2025-08-01', DATE '2025-08-07', 6);
```

```

INSERT INTO TradeFair VALUES (14, 'Agriculture Fair', 'Khulna', DATE '2025-02-20', DATE '2025-02-25', 5);
INSERT INTO TradeFair VALUES (15, 'Tourism Expo', 'CoxsBazar', DATE '2025-03-10', DATE '2025-03-14', 4);
INSERT INTO TradeFair VALUES (16, 'Energy Expo', 'Dhaka', DATE '2025-04-01', DATE '2025-04-06', 5);
INSERT INTO TradeFair VALUES (17, 'AI Expo', 'Dhaka', DATE '2025-05-15', DATE '2025-05-20', 5);
INSERT INTO TradeFair VALUES (18, 'Art Fair', 'Sylhet', DATE '2025-06-10', DATE '2025-06-15', 5);
INSERT INTO TradeFair VALUES (19, 'Gaming Expo', 'Dhaka', DATE '2025-07-15', DATE '2025-07-20', 5);
INSERT INTO TradeFair VALUES (20, 'Film Expo', 'Chattogram', DATE '2025-08-01', DATE '2025-08-07', 6);

select * from TradeFair

```

Results **Explain** **Describe** **Saved SQL** **History**

| FAIRID | TNAME | CITY | STARTDATE | ENDDATE | DURATION |
|--------|------------------|------------|-----------|-----------|----------|
| 1 | Tech Expo 2025 | Dhaka | 10-JAN-25 | 20-JAN-25 | 10 |
| 2 | Food Carnival | Chattogram | 05-FEB-25 | 12-FEB-25 | 7 |
| 3 | Book Fair | Dhaka | 01-MAR-25 | 15-MAR-25 | 14 |
| 4 | Startup Fair | Sylhet | 10-APR-25 | 15-APR-25 | 5 |
| 5 | Furniture Expo | Rajshahi | 05-MAY-25 | 10-MAY-25 | 5 |
| 6 | Car Show | Khulna | 01-JUN-25 | 07-JUN-25 | 6 |
| 8 | Education Expo | Chattogram | 15-AUG-25 | 20-AUG-25 | 5 |
| 9 | Clothing Fair | Dhaka | 01-SEP-25 | 05-SEP-25 | 4 |
| 10 | Sports Gear Fair | Sylhet | 01-OCT-25 | 08-OCT-25 | 7 |
| 11 | Toy Expo | Rajshahi | 01-NOV-25 | 05-NOV-25 | 4 |

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds [CSV Export](#)

Language: en-us Application Express 2.1.0.00.39 Copyright © 1999, 2006, Oracle. All rights reserved.

2) Hall Table Insertion :

```

INSERT INTO Hall VALUES (1, 'Hall A', 'Main Road', 'Expo Center', 1, 'A101', 1);

INSERT INTO Hall VALUES (2, 'Hall B', 'East Wing', 'Expo Center', 1, 'B102', 1);

INSERT INTO Hall VALUES (3, 'Hall C', 'West Wing', 'Expo Center', 2, 'C201', 2);

INSERT INTO Hall VALUES (4, 'Hall D', 'North Wing', 'Expo Center', 2, 'D202', 2);

INSERT INTO Hall VALUES (5, 'Hall E', 'South Wing', 'Expo Center', 3, 'E301', 3);

INSERT INTO Hall VALUES (6, 'Hall F', 'Main Block', 'Fair Building', 1, 'F101', 3);

INSERT INTO Hall VALUES (7, 'Hall G', 'Side Block', 'Fair Building', 1, 'G102', 4);

INSERT INTO Hall VALUES (8, 'Hall H', 'East Block', 'Fair Building', 2, 'H201', 4);

INSERT INTO Hall VALUES (9, 'Hall I', 'West Block', 'Trade Arena', 1, 'I101', 5);

INSERT INTO Hall VALUES (10, 'Hall J', 'North Block', 'Trade Arena', 2, 'J202', 5);

INSERT INTO Hall VALUES (11, 'Hall K', 'South Block', 'Trade Arena', 2, 'K203', 6);

INSERT INTO Hall VALUES (12, 'Hall L', 'Center Block', 'Trade Arena', 1, 'L101', 6);

INSERT INTO Hall VALUES (13, 'Hall M', 'North Hall', 'Mega Center', 1, 'M101', 7);

INSERT INTO Hall VALUES (14, 'Hall N', 'South Hall', 'Mega Center', 2, 'N201', 7);

INSERT INTO Hall VALUES (15, 'Hall O', 'East Hall', 'Mega Center', 3, 'O301', 8);

```

```

INSERT INTO Hall VALUES (16, 'Hall P', 'West Hall', 'Mega Center', 3, 'P302', 8);

INSERT INTO Hall VALUES (17, 'Hall Q', 'Main Plaza', 'Exhibition Hall', 1, 'Q101', 9);

INSERT INTO Hall VALUES (18, 'Hall R', 'Side Plaza', 'Exhibition Hall', 2, 'R202', 9);

INSERT INTO Hall VALUES (19, 'Hall S', 'Grand Hall', 'Exhibition Hall', 3, 'S301', 10);

INSERT INTO Hall VALUES (20, 'Hall T', 'Mini Hall', 'Exhibition Hall', 1, 'T101', 10);

```

The screenshot shows a browser window with the URL `127.0.0.1:8080/apex/r?p=4500:1003:258334110608334:NO::`. The title bar says "WhatsApp - Bayaan - Mera Musafir (Audio) - You - SQL Commands". The results section displays the following SQL code and its corresponding output table:

```

INSERT INTO Hall VALUES (3, 'Hall C', 'West Wing', 'Expo Center', 2, 'C201', 2);
INSERT INTO Hall VALUES (4, 'Hall D', 'North Wing', 'Expo Center', 2, 'D202', 2);
INSERT INTO Hall VALUES (5, 'Hall E', 'South Wing', 'Expo Center', 3, 'E301', 3);
INSERT INTO Hall VALUES (6, 'Hall F', 'Main Block', 'Fair Building', 1, 'F101', 3);
INSERT INTO Hall VALUES (7, 'Hall G', 'Side Block', 'Fair Building', 1, 'G102', 4);
INSERT INTO Hall VALUES (8, 'Hall H', 'East Block', 'Fair Building', 2, 'H201', 4);
INSERT INTO Hall VALUES (9, 'Hall I', 'West Block', 'Trade Arena', 1, 'I101', 5);
INSERT INTO Hall VALUES (10, 'Hall J', 'North Block', 'Trade Arena', 2, 'J202', 5);
INSERT INTO Hall VALUES (11, 'Hall K', 'South Block', 'Trade Arena', 2, 'K203', 6);
INSERT INTO Hall VALUES (12, 'Hall L', 'Center Block', 'Trade Arena', 1, 'L101', 6);
INSERT INTO Hall VALUES (13, 'Hall M', 'North Hall', 'Mega Center', 1, 'M101', 7);

```

Results

| HALLID | HNAME | LOCATION | BUILDINGNAME | FLOOR | APTNNO | FAIRID |
|--------|--------|-------------|---------------|-------|--------|--------|
| 1 | Hall A | Main Road | Expo Center | 1 | A101 | 1 |
| 2 | Hall B | East Wing | Expo Center | 1 | B102 | 1 |
| 3 | Hall C | West Wing | Expo Center | 2 | C201 | 2 |
| 4 | Hall D | North Wing | Expo Center | 2 | D202 | 2 |
| 5 | Hall E | South Wing | Expo Center | 3 | E301 | 3 |
| 6 | Hall F | Main Block | Fair Building | 1 | F101 | 3 |
| 7 | Hall G | Side Block | Fair Building | 1 | G102 | 4 |
| 8 | Hall H | East Block | Fair Building | 2 | H201 | 4 |
| 9 | Hall I | West Block | Trade Arena | 1 | I101 | 5 |
| 10 | Hall J | North Block | Trade Arena | 2 | J202 | 5 |

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds [CSV Export](#)

Application Express 2.1.0.0.39
Copyright © 1999, 2006, Oracle. All rights reserved.

Language: en-us

ENG INTL 9:37 PM 8/24/2025

3) Stall table insertion :

```

INSERT INTO Stall VALUES (1, 'Small', 5000, 1);

INSERT INTO Stall VALUES (2, 'Medium', 8000, 1);

INSERT INTO Stall VALUES (3, 'Large', 12000, 2);

INSERT INTO Stall VALUES (4, 'Small', 4000, 2);

INSERT INTO Stall VALUES (5, 'Medium', 7000, 3);

INSERT INTO Stall VALUES (6, 'Large', 15000, 3);

INSERT INTO Stall VALUES (7, 'Small', 4500, 4);

INSERT INTO Stall VALUES (8, 'Medium', 7500, 4);

INSERT INTO Stall VALUES (9, 'Large', 16000, 5);

```

```

INSERT INTO Stall VALUES (10, 'Small', 5000, 5);

INSERT INTO Stall VALUES (11, 'Medium', 8500, 6);

INSERT INTO Stall VALUES (12, 'Large', 17000, 6);

INSERT INTO Stall VALUES (13, 'Small', 6000, 7);

INSERT INTO Stall VALUES (14, 'Medium', 9000, 7);

INSERT INTO Stall VALUES (15, 'Large', 18000, 8);

INSERT INTO Stall VALUES (16, 'Small', 5500, 8);

INSERT INTO Stall VALUES (17, 'Medium', 9500, 9);

INSERT INTO Stall VALUES (18, 'Large', 20000, 9);

INSERT INTO Stall VALUES (19, 'Small', 7000, 10);

INSERT INTO Stall VALUES (20, 'Medium', 10000, 10);

```

The screenshot shows the Oracle SQL Developer interface. The top part displays the SQL command window with the following content:

```

INSERT INTO Stall VALUES (14, 'Medium', 9000, 7);
INSERT INTO Stall VALUES (15, 'Large', 18000, 8);
INSERT INTO Stall VALUES (16, 'Small', 5500, 8);
INSERT INTO Stall VALUES (17, 'Medium', 9500, 9);
INSERT INTO Stall VALUES (18, 'Large', 20000, 9);
INSERT INTO Stall VALUES (19, 'Small', 7000, 10);
INSERT INTO Stall VALUES (20, 'Medium', 10000, 10);

select * from Stall

```

The bottom part shows the results of the query, displaying a table with four columns: STALLID, SIZE, PRICE, and HALLID. The data is as follows:

| STALLID | SIZE | PRICE | HALLID |
|---------|--------|-------|--------|
| 1 | Small | 5000 | 1 |
| 2 | Medium | 8000 | 1 |
| 3 | Large | 12000 | 2 |
| 4 | Small | 4000 | 2 |
| 5 | Medium | 7000 | 3 |
| 6 | Large | 15000 | 3 |
| 7 | Small | 4500 | 4 |
| 8 | Medium | 7500 | 4 |
| 9 | Large | 16000 | 5 |
| 10 | Small | 5000 | 5 |

Below the table, a message indicates "More than 10 rows available. Increase rows selector to view more rows." The status bar at the bottom shows "10 rows returned in 0.00 seconds" and "CSV Export". The footer includes the application version "Application Express 2.1.0.00.39" and copyright information "Copyright © 1999, 2006, Oracle. All rights reserved".

4) Exhibitor table insertion :

```

INSERT INTO Exhibitor VALUES (1, 'TechCorp', 'Mr. Rahman', 'rahman@techcorp.com', '01711111111',
'rahman@techcorp.com');

```

```

INSERT INTO Exhibitor VALUES (2, 'Foodies', 'Mr. Karim', 'karim@foodies.com', '01722222222', 'karim@foodies.com');

```

```
INSERT INTO Exhibitor VALUES (3, 'BookHub', 'Ms. Ayesha', 'ayesha@bookhub.com', '01733333333',  
'ayesha@bookhub.com');  
  
INSERT INTO Exhibitor VALUES (4, 'StartUpX', 'Mr. Hasan', 'hasan@startupx.com', '01744444444',  
'hasan@startupx.com');  
  
INSERT INTO Exhibitor VALUES (5, 'FurniHouse', 'Mr. Anis', 'anis@furnihouse.com', '01755555555',  
'anis@furnihouse.com');  
  
INSERT INTO Exhibitor VALUES (6, 'AutoWorld', 'Mr. Alam', 'alam@autoworld.com', '01766666666',  
'alam@autoworld.com');  
  
INSERT INTO Exhibitor VALUES (7, 'MediTech', 'Dr. Rana', 'rana@meditech.com', '01777777777', 'rana@meditech.com');  
  
INSERT INTO Exhibitor VALUES (8, 'EduWorld', 'Mr. Bashir', 'bashir@eduworld.com', '01788888888',  
'bashir@eduworld.com');  
  
INSERT INTO Exhibitor VALUES (9, 'FashionFi', 'Ms. Jahan', 'jahan@fashionfi.com', '01799999999',  
'jahan@fashionfi.com');  
  
INSERT INTO Exhibitor VALUES (10, 'Sportify', 'Mr. Kabir', 'kabir@sportify.com', '01811111111', 'kabir@sportify.com');  
  
INSERT INTO Exhibitor VALUES (11, 'ToyLand', 'Mr. Amin', 'amin@toyland.com', '01822222222', 'amin@toyland.com');  
  
INSERT INTO Exhibitor VALUES (12, 'MobileHub', 'Mr. Imran', 'imran@mobilehub.com', '01833333333',  
'imran@mobilehub.com');  
  
INSERT INTO Exhibitor VALUES (13, 'ITSolutions', 'Ms. Sumi', 'sumi@itsolutions.com', '01844444444',  
'sumi@itsolutions.com');  
  
INSERT INTO Exhibitor VALUES (14, 'AgriWorld', 'Mr. Shuvo', 'shuvo@agriworld.com', '01855555555',  
'shuvo@agriworld.com');  
  
INSERT INTO Exhibitor VALUES (15, 'TravelNow', 'Ms. Rupa', 'rupa@travelnow.com', '01866666666',  
'rupa@travelnow.com');  
  
INSERT INTO Exhibitor VALUES (16, 'EnergyCo', 'Mr. Farid', 'farid@energyco.com', '01877777777',  
'farid@energyco.com');  
  
INSERT INTO Exhibitor VALUES (17, 'AIWorld', 'Mr. Saif', 'saif@aiworld.com', '01888888888', 'saif@aiworld.com');  
  
INSERT INTO Exhibitor VALUES (18, 'Artify', 'Ms. Sara', 'sara@artify.com', '01899999999', 'sara@artify.com');  
  
INSERT INTO Exhibitor VALUES (19, 'GameZone', 'Mr. Tanvir', 'tanvir@gamezone.com', '01911111111',  
'tanvir@gamezone.com');  
  
INSERT INTO Exhibitor VALUES (20, 'FilmHouse', 'Mr. Rony', 'rony@filmhouse.com', '01922222222',  
'rony@filmhouse.com');
```

```

SQL Commands
Home > SQL > SQL Commands
ORACLE Database Express Edition
User TRADEFAIR
Home Explain Describe Saved SQL History
Autocommit Display 10
INSERT INTO Exhibitor VALUES (9, 'FashionFi', 'Ms. Jahan', 'jahan@fashionfi.com', '01799999999', 'jahan@fashionfi.com');
INSERT INTO Exhibitor VALUES (10, 'Sportify', 'Mr. Kabir', 'kabir@sportify.com', '01811111111', 'kabir@sportify.com');
INSERT INTO Exhibitor VALUES (11, 'ToyLand', 'Mr. Amin', 'amin@toylad.com', '01822222222', 'amin@toylad.com');
INSERT INTO Exhibitor VALUES (12, 'MobileHub', 'Mr. Imran', 'imran@mobilehub.com', '01833333333', 'imran@mobilehub.com');
INSERT INTO Exhibitor VALUES (13, 'Foodies', 'Ms. Sumi', 'sumi@foodies.com', '01722222222', 'sumi@foodies.com');
INSERT INTO Exhibitor VALUES (14, 'AppWorld', 'Mr. Shuvo', 'shuvo@appworld.com', '01655555555', 'shuvo@appworld.com');
INSERT INTO Exhibitor VALUES (15, 'TravelNow', 'Ms. Rupa', 'rupa@travelnow.com', '01866666666', 'rupa@travelnow.com');
INSERT INTO Exhibitor VALUES (16, 'EnergyCo', 'Mr. Farid', 'farid@energyco.com', '01877777777', 'farid@energyco.com');
INSERT INTO Exhibitor VALUES (17, 'AutoWorld', 'Mr. Saif', 'saif@aworld.com', '01888888888', 'saif@aworld.com');
INSERT INTO Exhibitor VALUES (18, 'Artify', 'Ms. Sara', 'sara@artify.com', '01899999999', 'sara@artify.com');
INSERT INTO Exhibitor VALUES (19, 'GameZone', 'Mr. Tanvir', 'tanvir@gamezone.com', '01911111111', 'tanvir@gamezone.com');
INSERT INTO Exhibitor VALUES (20, 'FilmHouse', 'Mr. Romy', 'romy@filmhouse.com', '01922222222', 'romy@filmhouse.com');

select * from Exhibitor

```

Results Explain Describe Saved SQL History

| EXHIBITORID | ENAME | CONTACT | EMAIL | PHONE | EMAILADDRESS |
|-------------|-----------|------------|---------------------|-------------|---------------------|
| 1 | TechCorp | Mr. Rahman | rahman@techcorp.com | 01711111111 | rahman@techcorp.com |
| 2 | Foodies | Mr. Karim | karim@foodies.com | 01722222222 | karim@foodies.com |
| 3 | BookHub | Ms. Ayesha | ayesha@bookhub.com | 01733333333 | ayesha@bookhub.com |
| 4 | StartUpX | Mr. Hasan | hasan@startupp.com | 01744444444 | hasan@startupp.com |
| 5 | FurnHouse | Mr. Anis | anis@furnhouse.com | 01755555555 | anis@furnhouse.com |
| 6 | AutoWorld | Mr. Alam | alam@autoworld.com | 01766666666 | alam@autoworld.com |
| 7 | MediTech | Dr. Rana | rana@meditech.com | 01777777777 | rana@meditech.com |
| 8 | EduWorld | Mr. Bashir | bashir@eduwold.com | 01788888888 | bashir@eduwold.com |
| 9 | FashionFI | Ms. Jahan | jahan@fashionfi.com | 01799999999 | jahan@fashionfi.com |
| 10 | Sportify | Mr. Kabir | kabir@sportify.com | 01811111111 | kabir@sportify.com |

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds [CSV Export](#)

Application Express 2.1.0.0.39
Copyright © 1999-2005 Oracle. All rights reserved.

5) Salessummary Table insertion :

```

INSERT INTO SalesSummary VALUES (1, 1, 1, 120, 500000);

INSERT INTO SalesSummary VALUES (2, 2, 2, 200, 300000);

INSERT INTO SalesSummary VALUES (3, 3, 3, 150, 200000);

INSERT INTO SalesSummary VALUES (4, 4, 4, 80, 150000);

INSERT INTO SalesSummary VALUES (5, 5, 5, 60, 120000);

INSERT INTO SalesSummary VALUES (6, 6, 6, 40, 250000);

INSERT INTO SalesSummary VALUES (7, 7, 7, 300, 600000);

INSERT INTO SalesSummary VALUES (8, 8, 8, 210, 310000);

INSERT INTO SalesSummary VALUES (9, 9, 9, 400, 700000);

INSERT INTO SalesSummary VALUES (10, 10, 10, 500, 800000);

INSERT INTO SalesSummary VALUES (11, 11, 11, 50, 100000);

INSERT INTO SalesSummary VALUES (12, 12, 12, 80, 220000);

INSERT INTO SalesSummary VALUES (13, 13, 13, 100, 250000);

```

```

INSERT INTO SalesSummary VALUES (14, 14, 14, 70, 140000);

INSERT INTO SalesSummary VALUES (15, 15, 15, 90, 160000);

INSERT INTO SalesSummary VALUES (16, 16, 16, 110, 270000);

INSERT INTO SalesSummary VALUES (17, 17, 17, 200, 450000);

INSERT INTO SalesSummary VALUES (18, 18, 18, 130, 190000);

INSERT INTO SalesSummary VALUES (19, 19, 19, 300, 550000);

INSERT INTO SalesSummary VALUES (20, 20, 20, 220, 400000);

```

/

The screenshot shows the Oracle Database Express Edition interface. The SQL Commands tab contains the following SQL code:

```

INSERT INTO SalesSummary VALUES (7, 7, 7, 300, 600000);
INSERT INTO SalesSummary VALUES (8, 8, 8, 210, 310000);
INSERT INTO SalesSummary VALUES (9, 9, 9, 400, 700000);
INSERT INTO SalesSummary VALUES (10, 10, 10, 50, 100000);
INSERT INTO SalesSummary VALUES (11, 11, 11, 50, 120000);
INSERT INTO SalesSummary VALUES (12, 12, 12, 80, 220000);
INSERT INTO SalesSummary VALUES (13, 13, 13, 100, 250000);
INSERT INTO SalesSummary VALUES (14, 14, 14, 70, 140000);
INSERT INTO SalesSummary VALUES (15, 15, 15, 90, 160000);
INSERT INTO SalesSummary VALUES (16, 16, 16, 110, 270000);
INSERT INTO SalesSummary VALUES (17, 17, 17, 200, 450000);
INSERT INTO SalesSummary VALUES (18, 18, 18, 130, 190000);
INSERT INTO SalesSummary VALUES (19, 19, 19, 300, 550000);
INSERT INTO SalesSummary VALUES (20, 20, 20, 220, 400000);

```

The Results tab displays the data from the SalesSummary table:

| SUMMARYID | FAIRID | EXHIBITORID | NOOFPRODUCTSSOLD | TOTALSALES |
|-----------|--------|-------------|------------------|------------|
| 1 | 1 | 1 | 120 | 500000 |
| 2 | 2 | 2 | 200 | 300000 |
| 3 | 3 | 3 | 150 | 200000 |
| 4 | 4 | 4 | 80 | 150000 |
| 5 | 5 | 5 | 60 | 120000 |
| 6 | 6 | 6 | 40 | 250000 |
| 8 | 8 | 8 | 210 | 310000 |
| 9 | 9 | 9 | 400 | 700000 |
| 10 | 10 | 10 | 500 | 800000 |
| 11 | 11 | 11 | 50 | 100000 |

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds [CSV Export](#)

Application Express 2.1.0.0.39
Copyright © 1999-2025 Oracle All Rights Reserved

ENG INTL 10:02 PM 8/24/2025

6) Product table insertion :

```

INSERT INTO Product VALUES (1, 'Laptop Pro', 'Electronics', 'High performance laptop', 1200, 50, 1);

INSERT INTO Product VALUES (2, 'Burger Special', 'Food', 'Cheesy double burger', 5, 200, 2);

INSERT INTO Product VALUES (3, 'Novel X', 'Books', 'Bestseller novel', 10, 150, 3);

INSERT INTO Product VALUES (4, 'Startup Toolkit', 'Business', 'Entrepreneur starter kit', 150, 20, 4);

INSERT INTO Product VALUES (5, 'Wooden Chair', 'Furniture', 'Handmade wooden chair', 60, 80, 5);

INSERT INTO Product VALUES (6, 'SUV Car', 'Automobile', 'Luxury SUV', 25000, 5, 6);

INSERT INTO Product VALUES (7, 'X-Ray Machine', 'Medical', 'Portable X-Ray device', 5000, 10, 7);

```

```

INSERT INTO Product VALUES (8, 'Online Course', 'Education', 'Digital learning package', 50, 500, 8);

INSERT INTO Product VALUES (9, 'Designer Dress', 'Clothing', 'Trendy fashion dress', 40, 100, 9);

INSERT INTO Product VALUES (10, 'Football Shoes', 'Sports', 'Professional football boots', 120, 60, 10);

INSERT INTO Product VALUES (11, 'Toy Car', 'Toys', 'Battery operated toy car', 15, 150, 11);

INSERT INTO Product VALUES (12, 'Smartphone X', 'Electronics', 'Latest 5G phone', 999, 40, 12);

INSERT INTO Product VALUES (13, 'Server Software', 'IT', 'Enterprise software suite', 2000, 15, 13);

INSERT INTO Product VALUES (14, 'Tractor', 'Agriculture', 'High power tractor', 15000, 5, 14);

INSERT INTO Product VALUES (15, 'Tour Package', 'Travel', '7-day holiday package', 800, 25, 15);

INSERT INTO Product VALUES (16, 'Solar Panel', 'Energy', 'Renewable solar panel', 500, 100, 16);

INSERT INTO Product VALUES (17, 'AI Chip', 'AI Tech', 'Deep learning processor', 1500, 30, 17);

INSERT INTO Product VALUES (18, 'Painting Set', 'Art', 'Complete painting set', 70, 60, 18);

INSERT INTO Product VALUES (19, 'Gaming Console', 'Gaming', 'Next-gen gaming console', 450, 40, 19);

INSERT INTO Product VALUES (20, 'Film DVD', 'Entertainment', 'Blockbuster DVD collection', 20, 200, 20);

```

The screenshot shows the Oracle Database Express Edition interface. The SQL tab displays 20 INSERT statements for the Product table. The Results tab shows the inserted data in a grid format.

| PRODUCTID | PNAME | CATEGORY | DESCRIPTION | PRICE | STOCK | EXHIBITORID |
|-----------|-----------------|-------------|-----------------------------|-------|-------|-------------|
| 1 | Laptop Pro | Electronics | High performance laptop | 1200 | 50 | 1 |
| 2 | Burger Special | Food | Cheesy double burger | 5 | 200 | 2 |
| 3 | Novel X | Books | Bestseller novel | 10 | 150 | 3 |
| 4 | Startup Toolkit | Business | Entrepreneur starter kit | 150 | 20 | 4 |
| 5 | Wooden Chair | Furniture | Handmade wooden chair | 60 | 80 | 5 |
| 6 | SUV Car | Automobile | Luxury SUV | 25000 | 5 | 6 |
| 7 | X-Ray Machine | Medical | Portable X-Ray device | 5000 | 10 | 7 |
| 8 | Online Course | Education | Digital learning package | 50 | 500 | 8 |
| 9 | Designer Dress | Clothing | Trendy fashion dress | 40 | 100 | 9 |
| 10 | Football Shoes | Sports | Professional football boots | 120 | 60 | 10 |

7) Payment Table insertion:

```

INSERT INTO Payment VALUES (1, 5000, 'Card', 'TXN001', 'Completed', 1);

INSERT INTO Payment VALUES (2, 3000, 'Cash', 'TXN002', 'Completed', 2);

```

```
INSERT INTO Payment VALUES (3, 2000, 'Card', 'TXN003', 'Pending', 3);
INSERT INTO Payment VALUES (4, 1500, 'UPI', 'TXN004', 'Completed', 4);
INSERT INTO Payment VALUES (5, 1200, 'Card', 'TXN005', 'Completed', 5);
INSERT INTO Payment VALUES (6, 25000, 'Bank Transfer', 'TXN006', 'Completed', 6);
INSERT INTO Payment VALUES (7, 7000, 'Card', 'TXN007', 'Pending', 7);
INSERT INTO Payment VALUES (8, 3100, 'Cash', 'TXN008', 'Completed', 8);
INSERT INTO Payment VALUES (9, 4000, 'UPI', 'TXN009', 'Completed', 9);
INSERT INTO Payment VALUES (10, 8000, 'Card', 'TXN010', 'Completed', 10);
INSERT INTO Payment VALUES (11, 1000, 'Cash', 'TXN011', 'Completed', 11);
INSERT INTO Payment VALUES (12, 2200, 'UPI', 'TXN012', 'Completed', 12);
INSERT INTO Payment VALUES (13, 2500, 'Card', 'TXN013', 'Completed', 13);
INSERT INTO Payment VALUES (14, 1400, 'Cash', 'TXN014', 'Completed', 14);
INSERT INTO Payment VALUES (15, 1600, 'UPI', 'TXN015', 'Completed', 15);
INSERT INTO Payment VALUES (16, 2700, 'Card', 'TXN016', 'Pending', 16);
INSERT INTO Payment VALUES (17, 4500, 'Cash', 'TXN017', 'Completed', 17);
INSERT INTO Payment VALUES (18, 1900, 'UPI', 'TXN018', 'Completed', 18);
INSERT INTO Payment VALUES (19, 5500, 'Card', 'TXN019', 'Completed', 19);
INSERT INTO Payment VALUES (20, 4000, 'Bank Transfer', 'TXN020', 'Completed', 20);
```

The screenshot shows an Oracle Application Express interface for SQL Commands. The URL is 127.0.0.1:8080/apex/f?p=4500:1003:632537161326000::NO:::1. The page displays several INSERT statements into the Payment table:

```

INSERT INTO Payment VALUES (8, 3100, 'Cash', 'TXN008', 'Completed', 8);
INSERT INTO Payment VALUES (9, 4000, 'UPI', 'TXN009', 'Completed', 9);
INSERT INTO Payment VALUES (10, 8000, 'Card', 'TXN010', 'Completed', 10);
INSERT INTO Payment VALUES (11, 2000, 'Cash', 'TXN011', 'Completed', 11);
INSERT INTO Payment VALUES (12, 2200, 'UPI', 'TXN012', 'Completed', 12);
INSERT INTO Payment VALUES (13, 2500, 'Card', 'TXN013', 'Completed', 13);
INSERT INTO Payment VALUES (14, 1400, 'Cash', 'TXN014', 'Completed', 14);
INSERT INTO Payment VALUES (15, 1600, 'UPI', 'TXN015', 'Completed', 15);
INSERT INTO Payment VALUES (16, 2700, 'Card', 'TXN016', 'Pending', 16);
INSERT INTO Payment VALUES (17, 3000, 'Cash', 'TXN017', 'Completed', 17);
INSERT INTO Payment VALUES (18, 1900, 'UPI', 'TXN018', 'Completed', 18);
INSERT INTO Payment VALUES (19, 5500, 'Card', 'TXN019', 'Completed', 19);
INSERT INTO Payment VALUES (20, 4000, 'Bank Transfer', 'TXN020', 'Completed', 20);

```

Below the code, a SQL query `select * from Payment;` is run, and the results are displayed in a table:

| PAYMENTID | AMOUNT | MODE | TRANSACTIONID | STATUS | EXHIBITORID |
|-----------|--------|---------------|---------------|-----------|-------------|
| 1 | 5000 | Card | TXN001 | Completed | 1 |
| 2 | 3000 | Cash | TXN002 | Completed | 2 |
| 3 | 2000 | Card | TXN003 | Pending | 3 |
| 4 | 1500 | UPI | TXN004 | Completed | 4 |
| 5 | 1200 | Card | TXN005 | Completed | 5 |
| 6 | 25000 | Bank Transfer | TXN006 | Completed | 6 |
| 7 | 7000 | Card | TXN007 | Pending | 7 |
| 8 | 3100 | Cash | TXN008 | Completed | 8 |
| 9 | 4000 | UPI | TXN009 | Completed | 9 |
| 10 | 8000 | Card | TXN010 | Completed | 10 |

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds [CSV Export](#)

Language: en-us Application Express 2.1.0.00.39 Copyright © 1999, 2005, Oracle. All rights reserved.

8) Visitor table insertion :

INSERT INTO Visitor VALUES (1, 'Arif', 'Hossain', 'arif@demo.com', '01911111111', 'arif@demo.com', 'Technology');

INSERT INTO Visitor VALUES (2, 'Nusrat', 'Jahan', 'nusrat@demo.com', '01922222222', 'nusrat@demo.com', 'Food');

INSERT INTO Visitor VALUES (3, 'Sajid', 'Rahman', 'sajid@demo.com', '01933333333', 'sajid@demo.com', 'Books');

INSERT INTO Visitor VALUES (4, 'Mitu', 'Karim', 'mitu@demo.com', '01944444444', 'mitu@demo.com', 'Business');

INSERT INTO Visitor VALUES (5, 'Shamim', 'Anis', 'shamim@demo.com', '01955555555', 'shamim@demo.com', 'Furniture');

INSERT INTO Visitor VALUES (6, 'Alif', 'Alam', 'alif@demo.com', '01966666666', 'alif@demo.com', 'Automobiles');

INSERT INTO Visitor VALUES (7, 'Raisa', 'Akter', 'raisa@demo.com', '01977777777', 'raisa@demo.com', 'Medical');

INSERT INTO Visitor VALUES (8, 'Bashir', 'Ahmed', 'bashir@demo.com', '01988888888', 'bashir@demo.com', 'Education');

```
INSERT INTO Visitor VALUES (9, 'Jahan', 'Ara', 'jahan@demo.com', '01999999999', 'jahan@demo.com',  
'Fashion');  
  
INSERT INTO Visitor VALUES (10, 'Kabir', 'Hasan', 'kabir@demo.com', '01811111111', 'kabir@demo.com',  
'Sports');  
  
INSERT INTO Visitor VALUES (11, 'Amin', 'Chowdhury', 'amin@demo.com', '01822222222',  
'amin@demo.com', 'Toys');  
  
INSERT INTO Visitor VALUES (12, 'Imran', 'Rashid', 'imran@demo.com', '01833333333', 'imran@demo.com',  
'Mobiles');  
  
INSERT INTO Visitor VALUES (13, 'Sumi', 'Akter', 'sumi@demo.com', '01844444444', 'sumi@demo.com',  
'IT');  
  
INSERT INTO Visitor VALUES (14, 'Shuvo', 'Roy', 'shuvo@demo.com', '01855555555', 'shuvo@demo.com',  
'Agriculture');  
  
INSERT INTO Visitor VALUES (15, 'Rupa', 'Khan', 'rupa@demo.com', '01866666666', 'rupa@demo.com',  
'Travel');  
  
INSERT INTO Visitor VALUES (16, 'Farid', 'Uddin', 'farid@demo.com', '01877777777', 'farid@demo.com',  
'Energy');  
  
INSERT INTO Visitor VALUES (17, 'Saif', 'Hossain', 'saif@demo.com', '01888888888', 'saif@demo.com', 'AI');  
  
INSERT INTO Visitor VALUES (18, 'Sara', 'Begum', 'sara@demo.com', '01899999999', 'sara@demo.com',  
'Art');  
  
INSERT INTO Visitor VALUES (19, 'Tanvir', 'Hasan', 'tanvir@demo.com', '01711112222', 'tanvir@demo.com',  
'Gaming');  
  
INSERT INTO Visitor VALUES (20, 'Rony', 'Ahmed', 'rony@demo.com', '01722223333', 'rony@demo.com',  
'Film');
```

```

INSERT INTO Visitor VALUES (6, 'Alif', 'Alam', 'alif@demo.com', '019666666666', 'alif@demo.com', 'Automobile');
INSERT INTO Visitor VALUES (7, 'Raisa', 'Akter', 'raisa@demo.com', '019777777777', 'raisa@demo.com', 'Medical');
INSERT INTO Visitor VALUES (8, 'Bashir', 'Ahmed', 'bashir@demo.com', '019888888888', 'bashir@demo.com', 'Education');
INSERT INTO Visitor VALUES (9, 'Jahan', 'Ara', 'jahan@demo.com', '019999999999', 'jahan@demo.com', 'Fashion');
INSERT INTO Visitor VALUES (10, 'Kabir', 'Hasan', 'kabir@demo.com', '018111111111', 'kabir@demo.com', 'Sports');
INSERT INTO Visitor VALUES (11, 'Mitu', 'Karim', 'mitu@demo.com', '019222222222', 'mitu@demo.com', 'Books');
INSERT INTO Visitor VALUES (12, 'Tanjir', 'Rashid', 'tanjir@demo.com', '018333333333', 'tanjir@demo.com', 'Mobiles');
INSERT INTO Visitor VALUES (13, 'Sami', 'Akter', 'sumi@demo.com', '018444444444', 'sumi@demo.com', 'IT');
INSERT INTO Visitor VALUES (14, 'Shuvo', 'Roy', 'shuvo@demo.com', '018555555555', 'shuvo@demo.com', 'Agriculture');
INSERT INTO Visitor VALUES (15, 'Rupa', 'Khan', 'rupa@demo.com', '018666666665', 'rupa@demo.com', 'Travel');
INSERT INTO Visitor VALUES (16, 'Fahim', 'Hasan', 'fahim@demo.com', '018777777777', 'fahim@demo.com', 'Energy');
INSERT INTO Visitor VALUES (17, 'Saif', 'Hossain', 'saif@demo.com', '018888888888', 'saif@demo.com', 'AI');
INSERT INTO Visitor VALUES (18, 'Sara', 'Begum', 'sara@demo.com', '018999999999', 'sara@demo.com', 'Art');
INSERT INTO Visitor VALUES (19, 'Tanvir', 'Hasan', 'tanvir@demo.com', '017111122222', 'tanvir@demo.com', 'Gaming');
INSERT INTO Visitor VALUES (20, 'Rony', 'Ahmed', 'rony@demo.com', '017222233333', 'rony@demo.com', 'Film');

select * from Visitor

```

Results Explain Describe Saved SQL History

| VISITORID | FIRSTNAME | LASTNAME | CONTACT | PHONE | EMAILADDRESS | INTERESTS |
|-----------|-----------|----------|-----------------|--------------|-----------------|-------------|
| 1 | Arif | Hossain | arif@demo.com | 019111111111 | arif@demo.com | Technology |
| 2 | Nusrat | Jahan | nusrat@demo.com | 019222222222 | nusrat@demo.com | Food |
| 3 | Sajid | Rahman | sajid@demo.com | 019333333333 | sajid@demo.com | Books |
| 4 | Mitu | Karim | mitu@demo.com | 019444444444 | mitu@demo.com | Business |
| 5 | Shamim | Anis | shamim@demo.com | 019555555555 | shamim@demo.com | Furniture |
| 6 | Alif | Alam | alif@demo.com | 019666666666 | alif@demo.com | Automobiles |
| 7 | Raisa | Akter | raisa@demo.com | 019777777777 | raisa@demo.com | Medical |
| 8 | Bashir | Ahmed | bashir@demo.com | 019888888888 | bashir@demo.com | Education |
| 9 | Jahan | Ara | jahan@demo.com | 019999999999 | jahan@demo.com | Fashion |
| 10 | Kabir | Hasan | kabir@demo.com | 018111111111 | kabir@demo.com | Sports |

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds CSV Export

Application Express 2.1.0.0.39
Copyright © 1999-2006, Oracle. All rights reserved.

Language: en-us

ENG INTL 10:25 PM 8/24/2025

9) VisitorInterest table insertion :

```

INSERT INTO VisitorInterest VALUES (1, 'Tech Gadgets');

INSERT INTO VisitorInterest VALUES (2, 'Street Food');

INSERT INTO VisitorInterest VALUES (3, 'Fiction Books');

INSERT INTO VisitorInterest VALUES (4, 'Startups');

INSERT INTO VisitorInterest VALUES (5, 'Wood Furniture');

INSERT INTO VisitorInterest VALUES (6, 'Luxury Cars');

INSERT INTO VisitorInterest VALUES (7, 'Medical Devices');

INSERT INTO VisitorInterest VALUES (8, 'E-Learning');

INSERT INTO VisitorInterest VALUES (9, 'Fashion Wear');

INSERT INTO VisitorInterest VALUES (10, 'Football');

INSERT INTO VisitorInterest VALUES (11, 'Toys');

INSERT INTO VisitorInterest VALUES (12, 'Smartphones');

INSERT INTO VisitorInterest VALUES (13, 'IT Services');

INSERT INTO VisitorInterest VALUES (14, 'Farming Tools');

```

```

INSERT INTO VisitorInterest VALUES (15, 'Tourism');

INSERT INTO VisitorInterest VALUES (16, 'Solar Energy');

INSERT INTO VisitorInterest VALUES (17, 'AI Chips');

INSERT INTO VisitorInterest VALUES (18, 'Painting');

INSERT INTO VisitorInterest VALUES (19, 'Gaming Consoles');

INSERT INTO VisitorInterest VALUES (20, 'Movies');

```

/

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands tab, there is a code editor containing the following SQL statements:

```

INSERT INTO VisitorInterest VALUES (7, 'Medical Devices');
INSERT INTO VisitorInterest VALUES (8, 'E-learning');
INSERT INTO VisitorInterest VALUES (9, 'Fashion Wear');
INSERT INTO VisitorInterest VALUES (10, 'Football');
INSERT INTO VisitorInterest VALUES (11, 'Toys');
INSERT INTO VisitorInterest VALUES (12, 'Smartphones');
INSERT INTO VisitorInterest VALUES (13, 'IT Services');
INSERT INTO VisitorInterest VALUES (14, 'Medical Devices');
INSERT INTO VisitorInterest VALUES (15, 'Tourism');
INSERT INTO VisitorInterest VALUES (16, 'Solar Energy');
INSERT INTO VisitorInterest VALUES (17, 'AI Chips');
INSERT INTO VisitorInterest VALUES (18, 'Painting');
INSERT INTO VisitorInterest VALUES (19, 'Gaming Consoles');
INSERT INTO VisitorInterest VALUES (20, 'Movies')

```

Below the code editor, a results grid displays the data inserted into the VisitorInterest table:

| VISITORID | INTEREST |
|-----------|-----------------|
| 1 | Tech Gadgets |
| 2 | Street Food |
| 3 | Fiction Books |
| 4 | Startups |
| 5 | Wood Furniture |
| 6 | Luxury Cars |
| 7 | Medical Devices |
| 8 | E-learning |
| 9 | Fashion Wear |
| 10 | Football |

At the bottom of the results grid, it says "More than 10 rows available. Increase rows selector to view more rows." and "10 rows returned in 0.00 seconds".

The status bar at the bottom right shows "Application Express 2.1.0.0.39" and "Copyright © 1999-2006, Oracle. All rights reserved."

10) Ticket table insertion :

```

INSERT INTO Ticket VALUES (1, 'VIP', 500, DATE '2025-01-01', 1, 1);

INSERT INTO Ticket VALUES (2, 'Regular', 200, DATE '2025-02-01', 2, 2);

INSERT INTO Ticket VALUES (3, 'Student', 100, DATE '2025-03-01', 3, 3);

INSERT INTO Ticket VALUES (4, 'VIP', 500, DATE '2025-04-01', 4, 4);

INSERT INTO Ticket VALUES (5, 'Regular', 200, DATE '2025-05-01', 5, 5);

INSERT INTO Ticket VALUES (6, 'Student', 100, DATE '2025-06-01', 6, 6);

INSERT INTO Ticket VALUES (7, 'VIP', 500, DATE '2025-07-01', 7, 7);

```

```

INSERT INTO Ticket VALUES (8, 'Regular', 200, DATE '2025-08-01', 8, 8);

INSERT INTO Ticket VALUES (9, 'Student', 100, DATE '2025-09-01', 9, 9);

INSERT INTO Ticket VALUES (10, 'VIP', 500, DATE '2025-10-01', 10, 10);

INSERT INTO Ticket VALUES (11, 'Regular', 200, DATE '2025-11-01', 11, 11);

INSERT INTO Ticket VALUES (12, 'Student', 100, DATE '2025-12-01', 12, 12);

INSERT INTO Ticket VALUES (13, 'VIP', 500, DATE '2025-01-10', 13, 13);

INSERT INTO Ticket VALUES (14, 'Regular', 200, DATE '2025-02-10', 14, 14);

INSERT INTO Ticket VALUES (15, 'Student', 100, DATE '2025-03-10', 15, 15);

INSERT INTO Ticket VALUES (16, 'VIP', 500, DATE '2025-04-10', 16, 16);

INSERT INTO Ticket VALUES (17, 'Regular', 200, DATE '2025-05-10', 17, 17);

INSERT INTO Ticket VALUES (18, 'Student', 100, DATE '2025-06-10', 18, 18);

INSERT INTO Ticket VALUES (19, 'VIP', 500, DATE '2025-07-10', 19, 19);

INSERT INTO Ticket VALUES (20, 'Regular', 200, DATE '2025-08-10', 20, 20);

```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The SQL command entered is:

```

INSERT INTO Ticket VALUES (10, 'Student', 100, DATE '2025-08-01', 10, 10);

```

The results section displays the following data:

| TICKETID | TYPE | PRICE | PURCHASEDATE | FAIRID | VISITORID |
|----------|---------|-------|--------------|--------|-----------|
| 1 | VIP | 500 | 01-JAN-25 | 1 | 1 |
| 2 | Regular | 200 | 01-FEB-25 | 2 | 2 |
| 3 | Student | 100 | 01-MAR-25 | 3 | 3 |
| 4 | VIP | 500 | 01-APR-25 | 4 | 4 |
| 5 | Regular | 200 | 01-MAY-25 | 5 | 5 |
| 6 | Student | 100 | 01-JUN-25 | 6 | 6 |
| 8 | Regular | 200 | 01-AUG-25 | 8 | 8 |
| 9 | Student | 100 | 01-SEP-25 | 9 | 9 |
| 10 | VIP | 500 | 01-OCT-25 | 10 | 10 |
| 11 | Regular | 200 | 01-NOV-25 | 11 | 11 |

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds CSV Export

Language: en-us Application Express 2.1.0.0.39 Copyright © 1999-2006, Oracle. All rights reserved.

11) Feedback table insertion :

```
INSERT INTO Feedback VALUES (1, 1, 1, 5, 'Excellent fair');  
INSERT INTO Feedback VALUES (2, 2, 2, 4, 'Good experience');  
INSERT INTO Feedback VALUES (3, 3, 3, 5, 'Loved the books');  
INSERT INTO Feedback VALUES (4, 4, 4, 3, 'Average setup');  
INSERT INTO Feedback VALUES (5, 5, 5, 4, 'Nice furniture');  
INSERT INTO Feedback VALUES (6, 6, 6, 5, 'Awesome cars');  
INSERT INTO Feedback VALUES (7, 7, 7, 4, 'Medical tech useful');  
INSERT INTO Feedback VALUES (8, 8, 8, 3, 'Could improve');  
INSERT INTO Feedback VALUES (9, 9, 9, 5, 'Loved the clothes');  
INSERT INTO Feedback VALUES (10, 10, 10, 4, 'Good sports fair');  
INSERT INTO Feedback VALUES (11, 11, 11, 5, 'Kids enjoyed');  
INSERT INTO Feedback VALUES (12, 12, 12, 5, 'Great phone deals');  
INSERT INTO Feedback VALUES (13, 13, 13, 4, 'Informative');  
INSERT INTO Feedback VALUES (14, 14, 14, 5, 'Very useful for farmers');  
INSERT INTO Feedback VALUES (15, 15, 15, 4, 'Travel options were nice');  
INSERT INTO Feedback VALUES (16, 16, 16, 5, 'Energy solutions great');  
INSERT INTO Feedback VALUES (17, 17, 17, 5, 'AI sessions were excellent');  
INSERT INTO Feedback VALUES (18, 18, 18, 4, 'Nice art gallery');  
INSERT INTO Feedback VALUES (19, 19, 19, 5, 'Gaming was amazing');  
INSERT INTO Feedback VALUES (20, 20, 20, 4, 'Film expo was entertaining');
```

The screenshot shows the Oracle Database Express Edition interface. The SQL tab displays the following SQL code:

```

INSERT INTO Feedback VALUES (6, 6, 6, 5, 'Awesome cars');
INSERT INTO Feedback VALUES (7, 7, 7, 4, 'Medical tech useful');
INSERT INTO Feedback VALUES (8, 8, 8, 3, 'Could improve');
INSERT INTO Feedback VALUES (9, 9, 9, 5, 'Loved the clothes');
INSERT INTO Feedback VALUES (10, 10, 10, 4, 'Good sports fair');
INSERT INTO Feedback VALUES (11, 11, 11, 5, 'Nice art gallery');
INSERT INTO Feedback VALUES (12, 12, 12, 2, 'Great phone deals');
INSERT INTO Feedback VALUES (13, 13, 13, 4, 'Informative');
INSERT INTO Feedback VALUES (14, 14, 14, 5, 'Very useful for farmers');
INSERT INTO Feedback VALUES (15, 15, 15, 4, 'Travel options were nice');
INSERT INTO Feedback VALUES (16, 16, 16, 5, 'Food was delicious and great');
INSERT INTO Feedback VALUES (17, 17, 17, 3, 'Art sessions were excellent');
INSERT INTO Feedback VALUES (18, 18, 18, 4, 'Nice art gallery');
INSERT INTO Feedback VALUES (19, 19, 19, 5, 'Gaming was amazing');
INSERT INTO Feedback VALUES (20, 20, 20, 4, 'Film expo was entertaining');

select * from Feedback

```

The Results tab shows the following data in a table:

| FEEDBACKID | FAIRID | VISITORID | RATING | COMMENTS |
|------------|--------|-----------|--------|-------------------|
| 1 | 1 | 1 | 5 | Excellent fair |
| 2 | 2 | 2 | 4 | Good experience |
| 3 | 3 | 3 | 5 | Loved the books |
| 4 | 4 | 4 | 3 | Average setup |
| 5 | 5 | 5 | 4 | Nice furniture |
| 6 | 6 | 6 | 5 | Awesome cars |
| 8 | 8 | 8 | 3 | Could improve |
| 9 | 9 | 9 | 5 | Loved the clothes |
| 10 | 10 | 10 | 4 | Good sports fair |
| 11 | 11 | 11 | 5 | Kids enjoyed |

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds [CSV Export](#)

Language: en-US Application Express 2.1.0.0.39 Copyright © 1999-2006, Oracle. All rights reserved. 10:32 PM 8/24/2025 ENG INTL

Basic PL/SQL :

-- 1) Two Variables + 2 Operators

-- Project: International Trade Fair Management System

-- Semester: Summer 2024-2025

-- Course: Advanced Database Management System

-- Section: B, Group 6

DECLARE

v_price NUMBER;

v_qty NUMBER := 3;

v_total NUMBER;

BEGIN

SELECT Price INTO v_price FROM Stall WHERE StallID = 2; -- Stall 2 = Medium 8000

v_total := v_price * v_qty; -- Multiplication

```

DBMS_OUTPUT.PUT_LINE('Total Revenue (3 stalls) = ' || v_total);

v_total := v_total - 1000; -- Subtraction (discount)

DBMS_OUTPUT.PUT_LINE('After Discount = ' || v_total);

END;

```

The screenshot shows a browser window titled "SQL Commands" at the URL 127.0.0.1:8080/apex/f?p=4500:1003:3703836487.... The user is logged in as "TRADEFAIR". The page displays an SQL command editor and a results panel.

SQL Commands Editor:

```

-- Project: International Trade Fair Management System
-- Semester: Summer 2024-2025
-- Course: Advanced Database Management System
-- Section: B, Group 6
DECLARE
    v_price NUMBER;
    v_qty NUMBER := 3;
    v_total NUMBER;
BEGIN
    SELECT Price INTO v_price FROM Stall WHERE StallID = 2; -- Stall 2 =
Medium 8000
    v_total := v_price * v_qty; -- Multiplication

```

Results Panel:

```

Total Revenue (3 stalls) = 24000
After Discount = 23000

Statement processed.

0.01 seconds

```

At the bottom of the results panel, it says "Application Express 2.1.0.00.39" and "Copyright © 1999, 2006, Oracle. All rights reserved."

2)

-- 2) Two Single-row Functions
-- Project: International Trade Fair Management System
-- Semester: Summer 2024-2025
-- Course: Advanced Database Management System
-- Section: B, Group 6

DECLARE

v_name VARCHAR2(50);

v_date DATE;

BEGIN

SELECT FirstName, PurchaseDate
INTO v_name, v_date
FROM Visitor V JOIN Ticket T ON V.VisitorID = T.VisitorID
WHERE V.VisitorID = 1;

DBMS_OUTPUT.PUT_LINE('Visitor Name: ' || UPPER(v_name));

DBMS_OUTPUT.PUT_LINE('Ticket Date: ' || TO_CHAR(v_date, 'DD-MON-YYYY'));

END;

SQL Commands

User: TRADEFAIR

Home > SQL > SQL Commands

Autocommit Display 10

```
-- Project: International Trade Fair Management System
-- Semester: Summer 2024-2025
-- Course: Advanced Database Management System
-- Section: B, Group 6

DECLARE
    V_name VARCHAR2(50);
    V_date DATE;
BEGIN
    SELECT FirstName, PurchaseDate
    INTO V_name, V_date
    FROM Visitor V JOIN Ticket T ON V.VisitorID = T.VisitorID
    WHERE V.VisitorID = 1.
```

Results Explain Describe Saved SQL History

Visitor Name: ARIF
Ticket Date: 01-JAN-2025

Statement processed.

0.00 seconds

Application Express 2.1.0.00.39
Language: en-us Copyright © 1999, 2006, Oracle. All rights reserved.

10:48 PM 8/24/2025

3)

-- Project: International Trade Fair Management System
-- Semester: Summer 2024-2025
-- Course: Advanced Database Management System
-- Section: B, Group 6

DECLARE

v_count NUMBER;

v_max_price NUMBER;

BEGIN

SELECT COUNT(*), MAX(Price)

INTO v_count, v_max_price

FROM Product;

DBMS_OUTPUT.PUT_LINE('Total Exhibitors = ' || v_count);

DBMS_OUTPUT.PUT_LINE('Max Product Price = ' || v_max_price);

END;

/

The screenshot shows a browser window for Oracle Application Express (APEX) at the URL 127.0.0.1:8080/apex/f?p=4500:1003:3703836487... . The page title is "SQL Commands". The user is logged in as TRADEFAIR. The SQL editor contains the following PL/SQL block:

```
DECLARE
    v_count NUMBER;
    v_max_price NUMBER;
BEGIN
    SELECT COUNT(*), MAX(Price)
    INTO v_count, v_max_price
    FROM Product;

    DBMS_OUTPUT.PUT_LINE('Total Exhibitors = ' || v_count);
    DBMS_OUTPUT.PUT_LINE('Max Product Price = ' || v_max_price);
END;
```

The results section shows the output of the query:

```
Total Exhibitors = 20
Max Product Price = 25000

Statement processed.

0.00 seconds
```

The bottom status bar indicates the application version is 2.1.0.0.39, the language is en-us, and the copyright is Copyright © 1999, 2006, Oracle. All rights reserved.

```
-- 4) Two Loops (FOR & WHILE)

-- Project: International Trade Fair Management System

-- Semester: Summer 2024-2025

-- Course: Advanced Database Management System

-- Section: B, Group 6
```

```
DECLARE
```

```
    CURSOR cur_fair IS SELECT TName FROM TradeFair WHERE FairID <= 5;
```

```
    v_name VARCHAR2(100);
```

```
    i NUMBER := 1;
```

```
BEGIN
```

```
    -- FOR Loop
```

```
    FOR rec IN cur_fair LOOP
```

```
        DBMS_OUTPUT.PUT_LINE('Fair' || i || ':' || rec.TName);
```

```
        i := i + 1;
```

```
    END LOOP;
```

```
    -- WHILE loop with visitors
```

```
    i := 1;
```

```
    WHILE i <= 3 LOOP
```

```
        SELECT FirstName INTO v_name FROM Visitor WHERE VisitorID = i;
```

```
        DBMS_OUTPUT.PUT_LINE('Visitor' || i || ':' || v_name);
```

```
        i := i + 1;
```

```
    END LOOP;
```

```
END;
```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The user is logged in as TRADEFAIR. The SQL code entered is:

```
i := i + 1;
END LOOP;

-- WHILE loop with visitors
i := 1;
WHILE i <= 3 LOOP
    SELECT FirstName INTO v_name FROM Visitor WHERE VisitorID = i;
    DBMS_OUTPUT.PUT_LINE('Visitor ' || i || ': ' || v_name);
    i := i + 1;
END LOOP;
END;
```

The results pane displays the output of the SQL code:

```
Fair 1: Tech Expo 2025
Fair 2: Food Carnival
Fair 3: Book Fair
Fair 4: Startup Fair
Fair 5: Furniture Expo
Visitor 1: Arif
Visitor 2: Nusrat
Visitor 3: Sajid

Statement processed.
```

```

-- 5) Two Conditional Statements (IF & CASE)

-- Project: International Trade Fair Management System

-- Semester: Summer 2024-2025

-- Course: Advanced Database Management System

-- Section: B, Group 6

DECLARE

    v_rating NUMBER;

BEGIN

    SELECT Rating INTO v_rating FROM Feedback WHERE FeedbackID = 1; -- "Excellent fair"

    -- IF condition

    IF v_rating = 5 THEN

        DBMS_OUTPUT.PUT_LINE('Feedback is Excellent');

    ELSE

        DBMS_OUTPUT.PUT_LINE('Feedback is Not Excellent');

    END IF;

    -- CASE

    CASE

        WHEN v_rating >= 4 THEN DBMS_OUTPUT.PUT_LINE('Positive Feedback');

        WHEN v_rating = 3 THEN DBMS_OUTPUT.PUT_LINE('Neutral Feedback');

        ELSE DBMS_OUTPUT.PUT_LINE('Negative Feedback');

    END CASE;

END;
/

```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The user is logged in as TRADEFAIR. The SQL editor contains the following PL/SQL code:

```
-- Project: International Trade Fair Management System
-- Semester: Summer 2024-2025
-- Course: Advanced Database Management System
-- Section: B, Group 6
DECLARE
    v_rating NUMBER;
BEGIN
    SELECT Rating INTO v_rating FROM Feedback WHERE FeedbackID = 1; --
    "Excellent fair"
    --
    -- IF condition
    IF v_rating = 5 THEN
        DBMS_OUTPUT.PUT_LINE('Feedback is Excellent');
    END IF;
END;
```

The results pane shows the output of the executed code:

```
Feedback is Excellent
Positive Feedback

Statement processed.

0.00 seconds
```

At the bottom, the Application Express version is shown as 2.1.0.00.39, and the copyright notice is from 1999-2006.

-- 6) Two Subqueries
-- Project: International Trade Fair Management System
-- Semester: Summer 2024-2025
-- Course: Advanced Database Management System
-- Section: B, Group 6

```

-- Project: International Trade Fair Management System

DECLARE
    v_sales NUMBER;
    v_exhibitor VARCHAR2(100);

BEGIN
    -- Subquery 1: Max Sales
    SELECT MAX(TotalSales) INTO v_sales FROM SalesSummary;

    -- Subquery 2: Exhibitor who got it
    SELECT EName INTO v_exhibitor
    FROM Exhibitor
    WHERE ExhibitorID = (SELECT ExhibitorID FROM SalesSummary WHERE TotalSales = v_sales);

    DBMS_OUTPUT.PUT_LINE('Highest Sales = ' || v_sales || ' by ' || v_exhibitor);
END;

```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The user is logged in as TRADEFAIR. The SQL editor contains the following code:

```
-- Subquery 1: Max Sales
SELECT MAX(TotalSales) INTO v_sales FROM SalesSummary;

-- Subquery 2: Exhibitor who got it
SELECT EName INTO v_exhibitor
FROM Exhibitor
WHERE ExhibitorID = (SELECT ExhibitorID FROM SalesSummary WHERE
TotalSales = v_sales);

DBMS_OUTPUT.PUT_LINE('Highest Sales = ' || v_sales || ' by ' ||
v_exhibitor);
END;
```

The results section displays the output of the query:

```
Highest Sales = 800000 by Spotify
Statement processed.

0.02 seconds
```

At the bottom, the Application Express version is shown as 2.1.0.00.39, and the copyright notice is from 1999-2006.

7)

-- Project: International Trade Fair Management System
-- Semester: Summer 2024-2025
-- Course: Advanced Database Management System
-- Section: B, Group 6

-- Project: International Trade Fair Management System

-- Join 1: Fair and Hall

```
SELECT T.TName, H.HName
```

```
FROM TradeFair T
```

```
JOIN Hall H ON T.FairID = H.FairID
```

```
WHERE T.FairID <= 3;
```

-- Join 2: Exhibitor and Product

```
SELECT E.EName, P.PName, P.Price
```

```
FROM Exhibitor E
```

```
JOIN Product P ON E.ExhibitorID = P.ExhibitorID
```

```
WHERE P.Price > 500;
```

SQL Commands

127.0.0.1:8080/apex/f?p=4500:1003:3703836487...

Codeforces History YouTube OLD Competitive AI OTHER Other favorites

ORACLE Database Express Edition

User: TRADEFAIR

Home Logout Help

Home > SQL > SQL Commands

Autocommit Display 10

```
-- Join 1: Fair and Hall
SELECT T.TName, H.HName
FROM TradeFair T
JOIN Hall H ON T.FairID = H.FairID
WHERE T.FairID <= 3;

-- Join 2: Exhibitor and Product
SELECT E.EName, P.PName, P.Price
FROM Exhibitor E
JOIN Product P ON E.ExhibitorID = P.ExhibitorID
WHERE P.Price > 500;
```

Results Explain Describe Saved SQL History

| TNAME | HNAME |
|----------------|--------|
| Tech Expo 2025 | Hall A |
| Tech Expo 2025 | Hall B |
| Food Carnival | Hall C |
| Food Carnival | Hall D |
| Book Fair | Hall E |
| Book Fair | Hall F |

6 rows returned in 0.00 seconds [CSV Export](#)

Application Express 2.1.0.00.39
Copyright © 1999-2006 Oracle. All rights reserved.

Language English ENG INTL 10:56 PM 8/24/2025

/

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The user is logged in as TRADEFAIR. The interface includes a toolbar with various icons, a navigation bar with links like Home, Logout, and Help, and a main area for writing and executing SQL code.

SQL Code:

```
-- Join 1: Fair and Hall
SELECT T.TName, H.HName
FROM TradeFair T
JOIN Hall H ON T.FairID = H.FairID
WHERE T.FairID <= 3;

-- Join 2: Exhibitor and Product
SELECT E.EName, P.PName, P.Price
FROM Exhibitor E
JOIN Product P ON E.ExhibitorID = P.ExhibitorID
WHERE P.Price > 500;
```

Results:

| ENAME | PNAME | PRICE |
|-------------|-----------------|-------|
| TechCorp | Laptop Pro | 1200 |
| AutoWorld | SUV Car | 25000 |
| MediTech | X-Ray Machine | 5000 |
| MobileHub | Smartphone X | 999 |
| ITSolutions | Server Software | 2000 |
| AgriWorld | Tractor | 15000 |
| TravelNow | Tour Package | 800 |
| AIWorld | AI Chip | 1500 |

8 rows returned in 0.00 seconds CSV Export

System tray icons include: File, Database, Folder, Word, Excel, Powerpoint, and a redacted icon. System status: ENG INTL, WiFi, Battery, 10:56 PM, 8/24/2025.

Advance PL/SQL :

- a) Two Stored Functions
- Project: International Trade Fair Management System

```
-- Semester: Summer 2024-2025  
-- Course: Advanced Database Management System  
-- Section: B, Group 6
```

```
CREATE OR REPLACE FUNCTION TicketCount RETURN NUMBER IS
```

```
    v_count NUMBER;  
  
BEGIN  
  
    SELECT COUNT(*) INTO v_count FROM Ticket;  
  
    RETURN v_count;  
  
END;
```

```
CREATE OR REPLACE FUNCTION TotalRevenue RETURN NUMBER IS
```

```
    v_sum NUMBER;  
  
BEGIN  
  
    SELECT SUM(Amount) INTO v_sum FROM Payment;  
  
    RETURN v_sum;  
  
END;
```

SQL Commands

127.0.0.1:8080/apex/f?p=4500:1003:3703836487...

Codeforces History YouTube OLD Competitive AI OTHER Other favorites

ORACLE Database Express Edition

User: TRADEFAIR

Home > SQL > SQL Commands

Autocommit Display 10

```
CREATE OR REPLACE FUNCTION TicketCount RETURN NUMBER IS
    v_count NUMBER;
BEGIN
    SELECT COUNT(*) INTO v_count FROM Ticket;
    RETURN v_count;
END;
/

CREATE OR REPLACE FUNCTION TotalRevenue RETURN NUMBER IS
    v_sum NUMBER;
BEGIN
    SELECT SUM(Amount) INTO v_sum FROM Payment;
```

Results Explain Describe Saved SQL History

Function created.

0.03 seconds

Application Express 2.1.0.00.39

Language: en-us Copyright © 1999, 2006, Oracle. All rights reserved.

ENG INTL 11:00 PM 8/24/2025

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The user is logged in as TRADEFAIR. The SQL editor contains the following PL/SQL code:

```
SELECT COUNT(*) INTO v_count FROM ticket;
RETURN v_count;
END;
/
CREATE OR REPLACE FUNCTION TotalRevenue RETURN NUMBER IS
    v_sum NUMBER;
BEGIN
    SELECT SUM(Amount) INTO v_sum FROM Payment;
    RETURN v_sum;
END;
/
```

The code is displayed in a syntax-highlighted editor with red underlines indicating potential errors or warnings. The results section shows the message "Function created." and a execution time of "0.01 seconds". The bottom status bar indicates the application version is Application Express 2.1.0.00.39, the language is en-us, and the copyright notice is Copyright © 1999, 2006, Oracle. All rights reserved.

-- b) Two Stored Procedures

-- Project: International Trade Fair Management System

-- Semester: Summer 2024-2025

-- Course: Advanced Database Management System

-- Section: B, Group 6

```
CREATE OR REPLACE PROCEDURE ShowProduct(p_id NUMBER) IS
    v_name VARCHAR2(100);
    v_price NUMBER;
BEGIN
    SELECT PName, Price INTO v_name, v_price FROM Product WHERE ProductID = p_id;
    DBMS_OUTPUT.PUT_LINE('Product = ' || v_name || ', Price = ' || v_price);
END;
```

```
CREATE OR REPLACE PROCEDURE AddExhibitor(p_id NUMBER, p_name VARCHAR2, p_contact
VARCHAR2) IS
BEGIN
    INSERT INTO Exhibitor(ExhibitorID, EName, Contact)
    VALUES(p_id, p_name, p_contact);
    DBMS_OUTPUT.PUT_LINE('Exhibitor Added: ' || p_name);
END;
```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The user is logged in as TRADEFAIR. The SQL editor contains two PL/SQL procedures:

```
CREATE OR REPLACE PROCEDURE ShowProduct(p_id NUMBER) IS
    v_name VARCHAR2(100);
    v_price NUMBER;
BEGIN
    SELECT PName, Price INTO v_name, v_price FROM Product WHERE ProductID = p_id;
    DBMS_OUTPUT.PUT_LINE('Product = ' || v_name || ', Price = ' || v_price);
END;

CREATE OR REPLACE PROCEDURE AddExhibitor(p_id NUMBER, p_name VARCHAR2,
p_contact VARCHAR2) IS
```

The results section shows the message "Procedure created." and a execution time of "0.02 seconds".

At the bottom, the Application Express version is listed as 2.1.0.00.39, and the system status bar shows the language is en-us, the URL is 127.0.0.1:8080/apex/f?p=4500:1000:370383..., and the date and time are 8/24/2025 11:01 PM.

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The user is logged in as TRADEFAIR. The SQL editor contains the following PL/SQL code:

```
DBMS_OUTPUT.PUT_LINE('Product = ' || v_name || ', Price = ' || v_price);
END;

CREATE OR REPLACE PROCEDURE AddExhibitor(p_id NUMBER, p_name VARCHAR2,
p_contact VARCHAR2) IS
BEGIN
    INSERT INTO Exhibitor(ExhibitorID, EName, Contact)
    VALUES(p_id, p_name, p_contact);
    DBMS_OUTPUT.PUT_LINE('Exhibitor Added: ' || p_name);
END;
```

The code is highlighted with syntax coloring. The interface includes a toolbar with Autocommit checked, a display dropdown set to 10, and Save/Run buttons. Below the editor, there are tabs for Results, Explain, Describe, Saved SQL, and History. At the bottom, it shows "Procedure created.", "0.00 seconds", and the Application Express version 2.1.0.00.39. The status bar at the bottom right shows the date and time.

c)

-- c) Two Table-based Records

-- Project: International Trade Fair Management System

-- Semester: Summer 2024-2025

-- Course: Advanced Database Management System

-- Section: B, Group 6

```
DECLARE
    r_fair TradeFair%ROWTYPE;
    r_exh Exhibitor%ROWTYPE;
BEGIN
    SELECT * INTO r_fair FROM TradeFair WHERE FairID = 1;
    DBMS_OUTPUT.PUT_LINE('Fair: ' || r_fair.TName || ' in ' || r_fair.City);

    SELECT * INTO r_exh FROM Exhibitor WHERE ExhibitorID = 2;
    DBMS_OUTPUT.PUT_LINE('Exhibitor: ' || r_exh.EName || ', Contact: ' || r_exh.Contact);
END;
```

/

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The user is logged in as TRADEFAIR. The SQL editor contains the following PL/SQL code:

```
r_fair TradeFair%ROWTYPE;
r_exh Exhibitor%ROWTYPE;
BEGIN
  SELECT * INTO r_fair FROM TradeFair WHERE FairID = 1;
  DBMS_OUTPUT.PUT_LINE('Fair: ' || r_fair.TName || ' in ' || r_fair.City);

  SELECT * INTO r_exh FROM Exhibitor WHERE ExhibitorID = 2;
  DBMS_OUTPUT.PUT_LINE('Exhibitor: ' || r_exh.EName || ', Contact: ' || r_exh.Contact);
END;
```

The results pane displays the output of the DBMS_OUTPUT.PUT_LINE statements:

```
Fair: Tech Expo 2025 in Dhaka
Exhibitor: Foodies, Contact: Mr. Karim

Statement processed.
```

Execution time: 0.00 seconds

Application Express 2.1.0.00.39

Language: en-us Copyright © 1999, 2006, Oracle. All rights reserved.

ENG INTL 11:03 PM 8/24/2025

-- d) Two Explicit Cursors
-- Project: International Trade Fair Management System
-- Semester: Summer 2024-2025
-- Course: Advanced Database Management System

-- Section: B, Group 6

DECLARE

CURSOR cur_vis IS SELECT FirstName, Interests FROM Visitor;

v_name Visitor.FirstName%TYPE;

v_interest Visitor.Interests%TYPE;

BEGIN

OPEN cur_vis;

LOOP

FETCH cur_vis INTO v_name, v_interest;

EXIT WHEN cur_vis%NOTFOUND;

DBMS_OUTPUT.PUT_LINE('Visitor: ' || v_name || ' | Interest: ' || v_interest);

END LOOP;

CLOSE cur_vis;

END;

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The user is logged in as TRADEFAIR. The code entered is a PL/SQL block that opens a cursor, loops through it to print visitor names and interests, and then closes the cursor. The results section displays the output of the code execution.

```
Autocommit Display 10
BEGIN
    OPEN cur_vis;
    LOOP
        FETCH cur_vis INTO v_name, v_interest;
        EXIT WHEN cur_vis%NOTFOUND;
        DBMS_OUTPUT.PUT_LINE('Visitor: ' || v_name || ' | Interest: ' || v_interest);
    END LOOP;
    CLOSE cur_vis;
END;
```

Results

```
Visitor: Arif | Interest: Technology
Visitor: Nusrat | Interest: Food
Visitor: Sajid | Interest: Books
Visitor: Mitu | Interest: Business
Visitor: Shamim | Interest: Furniture
Visitor: Alif | Interest: Automobiles
Visitor: Raisa | Interest: Medical
Visitor: Bashir | Interest: Education
Visitor: Jahan | Interest: Fashion
Visitor: Kabir | Interest: Sports
Visitor: Amin | Interest: Toys
Visitor: Imran | Interest: Mobiles
```

-- e) Two Cursor-based Records

-- Project: International Trade Fair Management System

-- Semester: Summer 2024-2025

-- Course: Advanced Database Management System

-- Section: B, Group 6

```
DECLARE
  CURSOR cur_ex IS SELECT * FROM Exhibitor;
  rec cur_ex%ROWTYPE;
BEGIN
  OPEN cur_ex;
  LOOP
    FETCH cur_ex INTO rec;
    EXIT WHEN cur_ex%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE('Exhibitor: ' || rec.EName || ', Email: ' || rec.Email);
  END LOOP;
  CLOSE cur_ex;
END;
```

/

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The user is logged in as TRADEFAIR. The SQL editor contains the following PL/SQL block:

```
OPEN cur_ex;
LOOP
    FETCH cur_ex INTO rec;
    EXIT WHEN cur_ex%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE('Exhibitor: ' || rec.EName || ', Email: ' || rec.Email);
END LOOP;
CLOSE cur_ex;
END;
```

The results pane displays the output of the PL/SQL block, listing ten exhibitors with their names and emails:

- Exhibitor: TechCorp, Email: rahman@techcorp.com
- Exhibitor: Foodies, Email: karim@foodies.com
- Exhibitor: BookHub, Email: ayesha@bookhub.com
- Exhibitor: StartUpX, Email: hasan@startupx.com
- Exhibitor: FurniHouse, Email: anis@furnihouse.com
- Exhibitor: AutoWorld, Email: alam@autoworld.com
- Exhibitor: MediTech, Email: rana@meditech.com
- Exhibitor: EduWorld, Email: bashir@eduworld.com
- Exhibitor: FashionFi, Email: jahan@fashionfi.com
- Exhibitor: Sportify, Email: kabir@sportify.com
- Exhibitor: ToyLand, Email: amin@toyladn.com
- Exhibitor: MobileHub, Email: imran@mobilehub.com

The interface includes standard browser controls, a navigation bar (Home, Logout, Help), and a toolbar with various icons.

-- f) Two Row-level Triggers

-- Project: International Trade Fair Management System

-- Semester: Summer 2024-2025

-- Course: Advanced Database Management System

-- Section: B, Group 6

```
CREATE OR REPLACE TRIGGER trg_after_insert_payment
AFTER INSERT ON Payment
FOR EACH ROW
BEGIN
    DBMS_OUTPUT.PUT_LINE('Payment received: ' || :NEW.Amount || ' for Exhibitor' || :NEW.ExhibitorID);
END;
```

```
CREATE OR REPLACE TRIGGER trg_after_update_product_stock
AFTER UPDATE OF Stock ON Product
FOR EACH ROW
BEGIN
    DBMS_OUTPUT.PUT_LINE('Stock updated for ' || :NEW.PName || ', New Stock = ' || :NEW.Stock);
END;
```

SQL Commands

127.0.0.1:8080/apex/f?p=4500:1003:3703836487...

Codeforces History YouTube OLD Competitive AI OTHER Other favorites

ORACLE Database Express Edition

User: TRADEFAIR

Home > SQL > SQL Commands

Autocommit Display 10 Save Run

```
CREATE OR REPLACE TRIGGER trg_after_insert_payment
AFTER INSERT ON Payment
FOR EACH ROW
BEGIN
    DBMS_OUTPUT.PUT_LINE('Payment received: ' || :NEW.Amount || ' for
Exhibitor ' || :NEW.ExhibitorID);
END;

CREATE OR REPLACE TRIGGER trg_after_update_product_stock
AFTER UPDATE OF Stock ON Product
FOR EACH ROW
BEGIN
```

Results Explain Describe Saved SQL History

Trigger created.

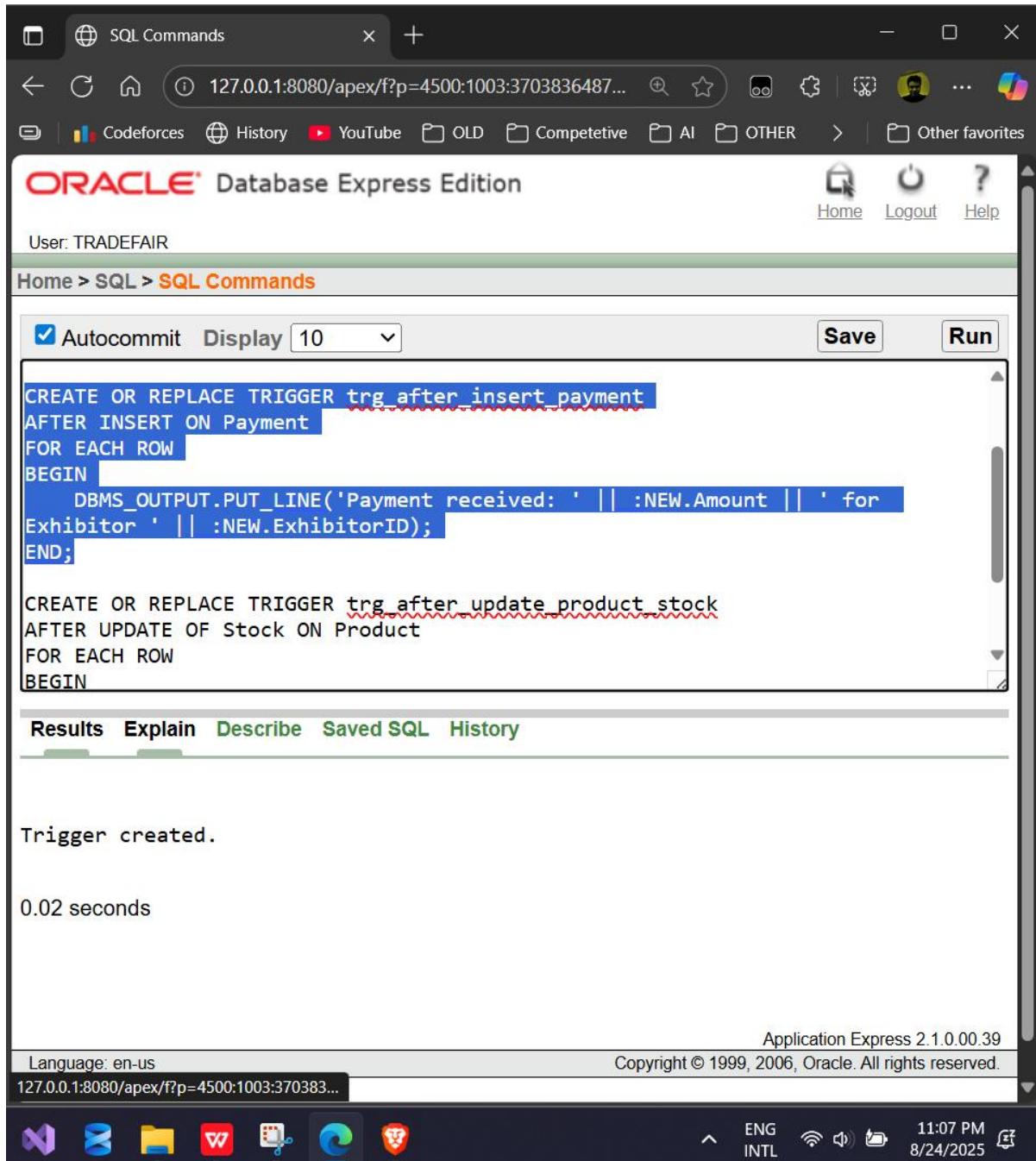
0.02 seconds

Application Express 2.1.0.00.39

Language: en-us Copyright © 1999, 2006, Oracle. All rights reserved.

127.0.0.1:8080/apex/f?p=4500:1003:370383...

ENG INTL 11:07 PM 8/24/2025



SQL Commands

127.0.0.1:8080/apex/f?p=4500:1003:3703836487...

Codeforces History YouTube OLD Competitive AI OTHER Other favorites

ORACLE Database Express Edition

User: TRADEFAIR

Home > SQL > SQL Commands

Autocommit Display 10 Save Run

```
DBMS_OUTPUT.PUT_LINE('Payment received: ' || :NEW.Amount || ' for
Exhibitor ' || :NEW.ExhibitorID);
END;
```

```
CREATE OR REPLACE TRIGGER trg_after_update_product_stock
AFTER UPDATE OF Stock ON Product
FOR EACH ROW
BEGIN
    DBMS_OUTPUT.PUT_LINE('Stock updated for ' || :NEW.PName || ', New Stock =
' || :NEW.Stock);
END;
```

Results Explain Describe Saved SQL History

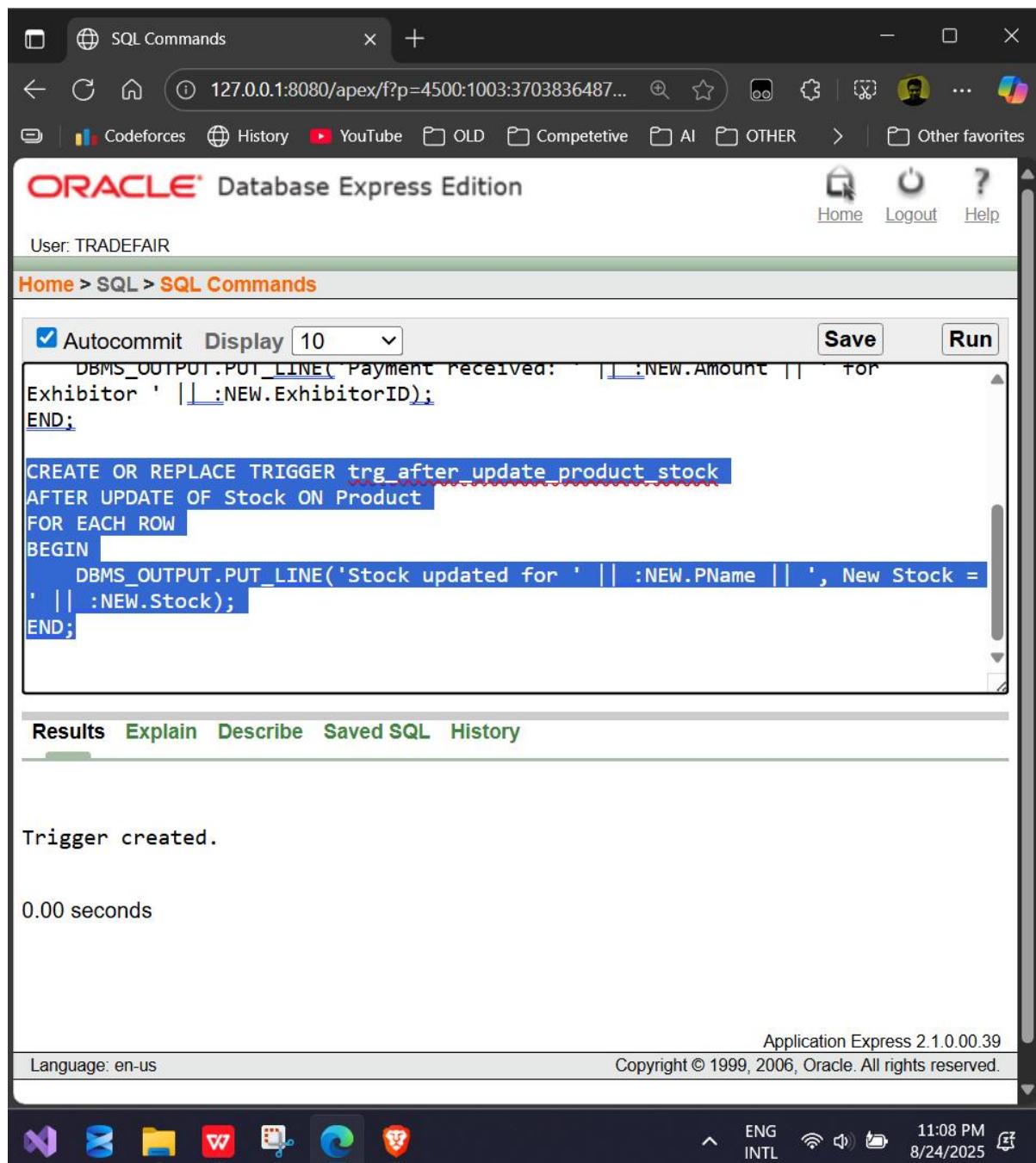
Trigger created.

0.00 seconds

Application Express 2.1.0.00.39

Language: en-us Copyright © 1999, 2006, Oracle. All rights reserved.

ENG INTL 11:08 PM 8/24/2025



-- g) Two Statement-level Triggers
-- Project: International Trade Fair Management System
-- Semester: Summer 2024-2025
-- Course: Advanced Database Management System
-- Section: B, Group 6

```
CREATE OR REPLACE TRIGGER trg_before_insert_ticket
BEFORE INSERT ON Ticket
BEGIN
    DBMS_OUTPUT.PUT_LINE('About to insert Ticket...');

END;
```

```
CREATE OR REPLACE TRIGGER trg_after_delete_feedback
AFTER DELETE ON Feedback
BEGIN
    DBMS_OUTPUT.PUT_LINE('Feedback record deleted.');

END;
```

SQL Commands

127.0.0.1:8080/apex/f?p=4500:1003:3703836487...

Codeforces History YouTube OLD Competitive AI OTHER Other favorites

ORACLE Database Express Edition

User: TRADEFAIR

Home Logout Help

Home > SQL > SQL Commands

Autocommit Display 10

```
-- Project: International Trade Fair Management System
-- Semester: Summer 2024-2025
-- Course: Advanced Database Management System
-- Section: B, Group 6

CREATE OR REPLACE TRIGGER trg_before_insert_ticket
BEFORE INSERT ON Ticket
BEGIN
    DBMS_OUTPUT.PUT_LINE('About to insert Ticket...');

END;
```

Results Explain Describe Saved SQL History

Trigger created.

0.00 seconds

Application Express 2.1.0.00.39

Language: en-us Copyright © 1999, 2006, Oracle. All rights reserved.

ENG INTL 11:09 PM 8/24/2025

The screenshot shows a browser window for Oracle Application Express (APEX) with the URL `127.0.0.1:8080/apex/f?p=4500:1003:3703836487...`. The page title is "SQL Commands". The user is "TRADEFAIR". The navigation bar includes links for Home, Logout, and Help. The main content area shows SQL code in a code editor:

```
Autocommit Display 10 Save Run
BEGIN
  DBMS_OUTPUT.PUT_LINE('About to insert Ticket...');

END;

CREATE OR REPLACE TRIGGER trg_after_delete_feedback
AFTER DELETE ON Feedback
BEGIN
  DBMS_OUTPUT.PUT_LINE('Feedback record deleted.');
END;
```

Below the code editor, there are tabs for Results, Explain, Describe, Saved SQL, and History. The Results tab is selected, displaying the output:

Trigger created.
0.00 seconds

At the bottom, the footer information includes "Application Express 2.1.0.00.39", "Language: en-us", "Copyright © 1999, 2006, Oracle. All rights reserved.", and system status icons.

-- g) Two Packages

-- Project: International Trade Fair Management System

-- Semester: Summer 2024-2025

```
-- Course: Advanced Database Management System
-- Section: B, Group 6

CREATE OR REPLACE PACKAGE TradeFairPkg AS

    PROCEDURE ShowFairInfo(p_id NUMBER);

    FUNCTION GetExhibitorName(p_id NUMBER) RETURN VARCHAR2;

END TradeFairPkg;
```

```
CREATE OR REPLACE PACKAGE BODY TradeFairPkg AS

    PROCEDURE ShowFairInfo(p_id NUMBER) IS
        v_name VARCHAR2(100);
        v_city VARCHAR2(50);

    BEGIN
        SELECT TName, City INTO v_name, v_city FROM TradeFair WHERE FairID = p_id;
        DBMS_OUTPUT.PUT_LINE('Fair: ' || v_name || ' in ' || v_city);
    END;
```

```
FUNCTION GetExhibitorName(p_id NUMBER) RETURN VARCHAR2 IS
    v_ename VARCHAR2(100);

BEGIN
    SELECT EName INTO v_ename FROM Exhibitor WHERE ExhibitorID = p_id;
    RETURN v_ename;
END;

END TradeFairPkg;
```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The user is logged in as TRADEFAIR. The SQL editor contains the following PL/SQL code:

```
-- Project: International Trade Fair Management System
-- Semester: Summer 2024-2025
-- Course: Advanced Database Management System
-- Section: B, Group 6
CREATE OR REPLACE PACKAGE TradeFairPkg AS
    PROCEDURE ShowFairInfo(p_id NUMBER);
    FUNCTION GetExhibitorName(p_id NUMBER) RETURN VARCHAR2;
END TradeFairPkg;

CREATE OR REPLACE PACKAGE BODY TradeFairPkg AS
    PROCEDURE ShowFairInfo(p_id NUMBER) IS
        v_name VARCHAR2(100);
    BEGIN
        -- Implementation of ShowFairInfo procedure
    END;
    -- Implementation of GetExhibitorName function
END TradeFairPkg;
```

The code is highlighted with red underlines, indicating syntax errors or warnings. The interface includes a toolbar with Autocommit checked, a display dropdown set to 10, and buttons for Save and Run. Below the code, there are tabs for Results, Explain, Describe, Saved SQL, and History. The results pane shows the message "Package created." and a execution time of "0.02 seconds". The bottom status bar shows the application version as Application Express 2.1.0.00.39, the language as en-us, and the copyright notice from Oracle. The system tray at the bottom right shows the date and time as 8/24/2025 11:13 PM.

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The user is logged in as TRADEFAIR. The SQL editor contains the following PL/SQL code:

```
-- Project: International Trade Fair Management System
-- Semester: Summer 2024-2025
-- Course: Advanced Database Management System
-- Section: B, Group 6
CREATE OR REPLACE PACKAGE TradeFairPkg AS
    PROCEDURE ShowFairInfo(p_id NUMBER);
    FUNCTION GetExhibitorName(p_id NUMBER) RETURN VARCHAR2;
END TradeFairPkg;

CREATE OR REPLACE PACKAGE BODY TradeFairPkg AS
    PROCEDURE ShowFairInfo(p_id NUMBER) IS
        V_NAME VARCHAR2(100);
    BEGIN
        SELECT EXHIBITOR_NAME INTO V_NAME
        FROM EXHIBITOR
        WHERE EXHIBITOR_ID = p_id;
        DBMS_OUTPUT.PUT_LINE('Exhibitor Name: ' || V_NAME);
    END;
END;
```

The results section shows the message "Package Body created." and a execution time of "0.00 seconds".

At the bottom, the application version is Application Express 2.1.0.00.39, the language is en-us, and the copyright notice is Copyright © 1999, 2006, Oracle. All rights reserved.

Conclusion

In this project, we successfully designed and implemented the International Trade Fair Management System using Oracle 10g. The system manages trade fair information such as fairs, halls, exhibitors, products, visitors, tickets, and payments. We also applied both Basic PL/SQL and Advanced PL/SQL concepts like functions, procedures, triggers, cursors, and packages, which helped us to strengthen our practical knowledge of database programming.

Our findings show that the system can handle fair management tasks efficiently, reduce manual work, and provide useful analytical results such as sales summaries and visitor details. The project demonstrates how database management systems can be applied in real-world scenarios for automation and better decision-making.

Future Work

- In the future, we plan to improve this project by:
- Developing a web-based interface for easier access and management.
- Adding real-time reporting and dashboards for exhibitors and organizers.
- Implementing security features like role-based access control for data safety.
- Integrating online payment gateways to make the system more practical.
- Extending the system with AI-based analytics to predict visitor interest and sales trends.

These improvements will make the system more user-friendly, scalable, and suitable for real trade fair operations.