

GEBZE TECHNICAL UNIVERSITY

CSE 414

**Introduction To Databases
Term Project**

**TEKNOFEST
MANAGEMENT SYSTEM**

**PROJECT REPORT BY
Asuman Sare ERGÜT
1901042657**

TABLE OF CONTENTS

TABLE OF CONTENTS	2
1) PROBLEM DEFINITION	4
2) SETUP ENVIRONMENT	4
3) USER REQUIREMENTS	5
Organizers	5
Juries	5
Competitors	5
Visitors	5
4) TABLES	5
5) ER DIAGRAM	6
6) FUNCTIONAL DEPENDENCIES	7
• Competition	7
• Organizer	7
• Venue	7
• Competitor	7
• Ticket	7
• Visitor	7
• Jury	7
• Report	7
7) NORMALIZATIONS	8
• Venue Table	8
• Report Table	8
• Competitor Table	9
• Organizer Table	9
• Visitor Table	9
• Jury Table	9
• Ticket Table	9
• Competition Table	10
8) QUERIES	10
• SELECT	11
• SELECT * FROM Competition	11
• SELECT * FROM Jury	11
• SELECT * FROM Venue	11
• INSERT	12
• INSERT INTO Competition (From Organizer View)	12
• INSERT INTO Jury (From Jury View)	13
• INSERT INTO Competitor (From Competitor View)	14
• DELETE	15
• DELETE id FROM Competition	15
• DELETE ticket_id FROM Ticket	16
• DELETE id FROM Visitor	17
• UPDATE	18
• UPDATE Competition	18

• UPDATE Jury	19
• UPDATE Visitor	20
9) JOIN	21
• LEFT JOIN	21
• RIGHT JOIN	21
• FULL JOIN	21
10) TRIGGERS	22
• Trigger 1: Venue Capacity - Ticket Quantity	22
• Trigger 2: New Competition Date	22
• Trigger 3: Near Future Competition	23
• Trigger 4: Ticket Quantity	24
• Trigger 5: Visitor Age	24
12) VIEWS	25
• View 1: Competitor View	25
• View 2: Visitor View	26
• View 3: Jury View	26
• View 4: Jury Info View	26
• View 5: Ticket View	27
13) TRANSACTIONS	27
• Transaction 1: Volunteering	27
• Transaction 2: Buying Ticket	27
• Transaction 3: Canceling Ticket	28
• Transaction 4: Attending Competition - Jury	28
14) PROCEDURES	29
• Procedure 1: AddReport	29
• Procedure 2: AddVenue	29
• Procedure 3: AddOrganizer	29

1) PROBLEM DEFINITION

The main purpose of this project is to efficiently manage the database of Teknofest Organization. This aim will be achieved by creating a comprehensive database that includes information about competitions, event organizers, competitors, juries, venues that competitions occur and other relevant entities.

The Teknofest Management System will enable organizers to minimize their time and resources by systematically maintaining competition details, allowing for quick access to the data when needed. By using this system, competition organizers can reduce their workload, save costs, and minimize the use of physical documents. The user-friendly interface of the Teknofest management system will greatly assist organizers in making their process easier and the system more efficient.

2) SETUP ENVIRONMENT

First I've installed the Tkinter for interface, and mysql as sql server.

How to install:

Since I'm using Ubuntu 22.04 and Python 3.11, installation may change according to different versions. Here are mines:

For Tkinter:

- `sudo apt-get install python3-tk`
- You can verify the installation with: `python3 -m tkinter`

For MySQL:

- `sudo apt update`
- `sudo apt-get install mysql-server`
- To fix Access denied for user 'root'@'localhost'
- `sudo mysql -u root -p`
- `show databases;`
- `use mysql`
- `update user set plugin='mysql_native_password' where user='root';`
- `flush privileges;`
- Then finally Type :
- `mysql -u root -p`

Also for Tkinter-MySQL connection:

- `pip install mysql-connector-python`

3) USER REQUIREMENTS

Teknofest Management System has 4 different main users.

Organizers

- Can create, update, cancel competition,
- Can see competitions and details,
- Can generate reports on competition,
- Can view reports,
- Can view juries,
- Can view venues

Juries

- Can attend competitions,
- Can see competitions,
- Can change their expertise,
- Can list other juries,
- Can add jury

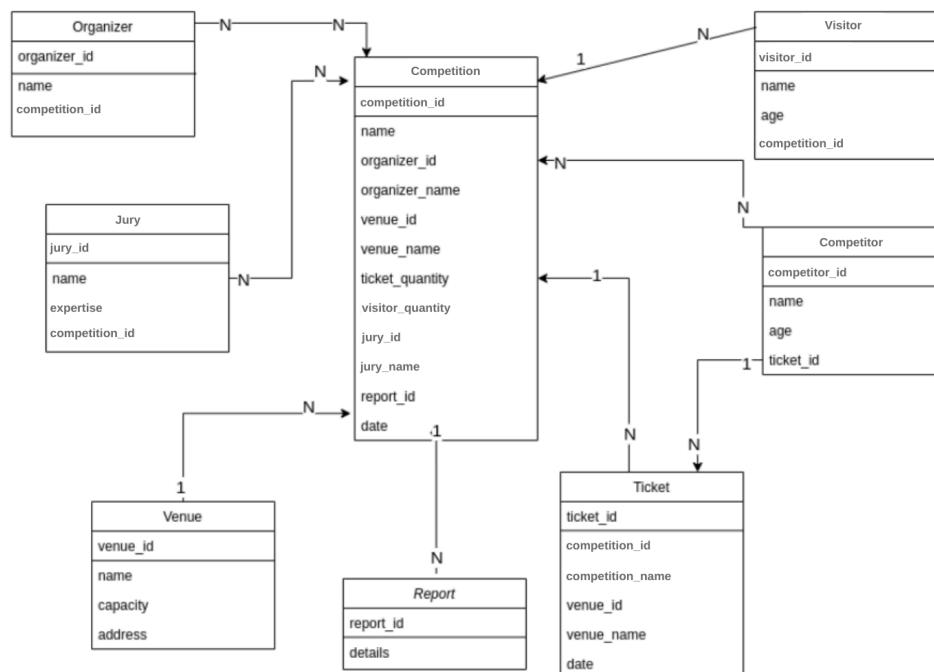
Competitors

- Can see competitions,
- Can list and add competitors,
- Can list tickets,
- Can buy and cancel ticket.

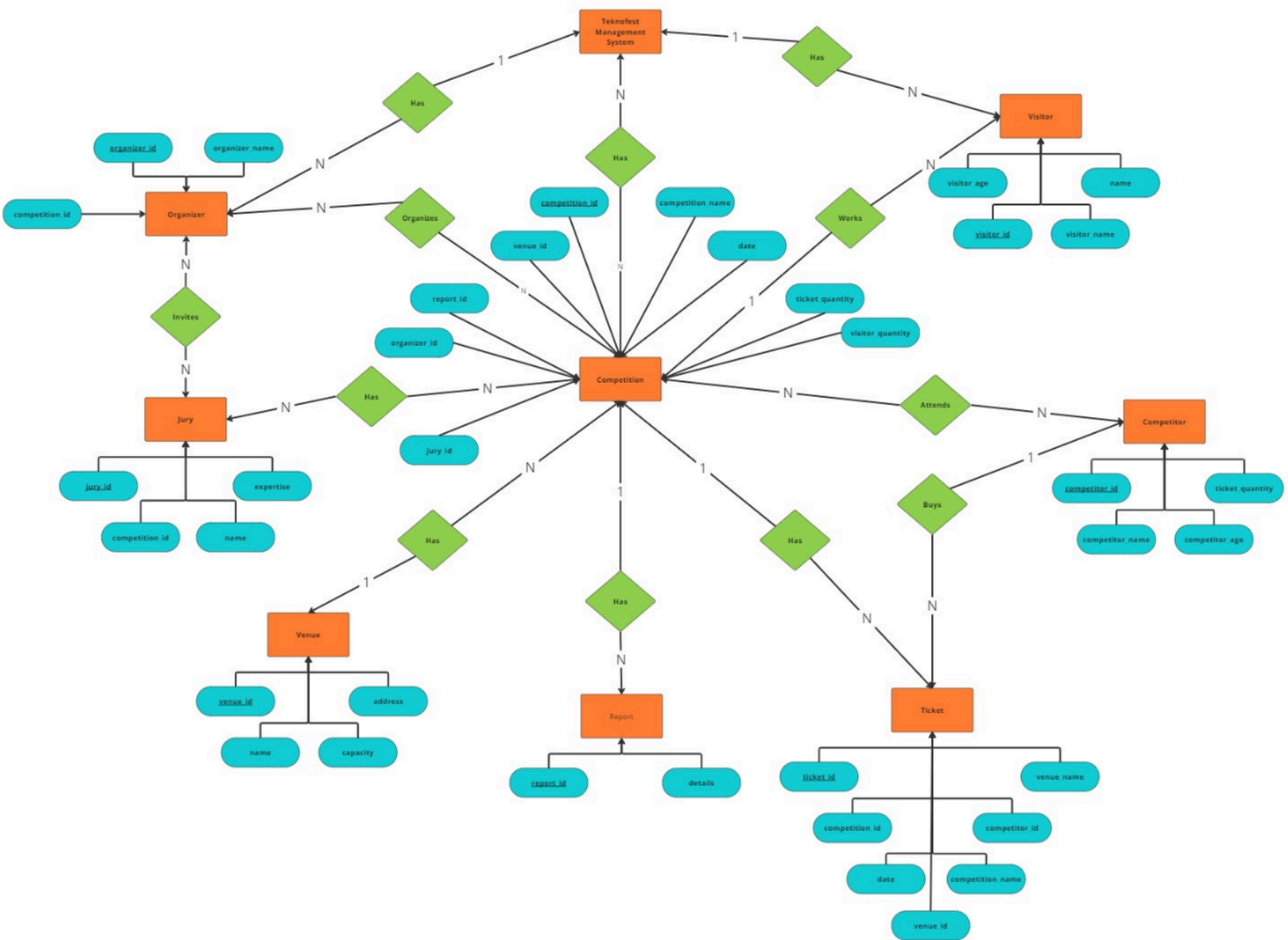
Visitors

- Can see competitions and details,
- Can volunteer,
- Can list juries,
- Can add,edit and remove visitor.

4) TABLES



5) ER DIAGRAM



6) FUNCTIONAL DEPENDENCIES

- **Competition**

competition_id → competition_name, **venue_id**, venue_name, **organizer_id**, organizer_name, **jury_id**, jury_name, ticket_quantity, visitor_quantity, **report_id**, date

- **Organizer**

organizer_id → organizer_name, **competition_id**

- **Venue**

venue_id → venue_name, capacity, address

- **Competitor**

competitor_id → competitor_name, competitor_age, **ticket_id**

- **Ticket**

ticket_id → **competition_id**, competition_name, date, **venue_id**, venue_name

- **Visitor**

visitor_id → visitor_name, visitor_age, **competition_id**

- **Jury**

jury_id → jury_name, expertise, **competition_id**

- **Report**

report_id → details

7) NORMALIZATIONS

Boyce-Codd Normal Form

- A relation schema R is in BCNF with respect to a set F of functional dependencies if for all functional dependencies in F^+ of the form

$$\alpha \rightarrow \beta$$

where $\alpha \subseteq R$ and $\beta \subseteq R$, at least one of the following holds:

- $\alpha \rightarrow \beta$ is trivial (i.e., $\beta \subseteq \alpha$)
- α is a superkey for R

Third Normal Form

- A relation schema R is in **third normal form (3NF)** if for all:

$$\alpha \rightarrow \beta \text{ in } F^+$$

at least one of the following holds:

- $\alpha \rightarrow \beta$ is trivial (i.e., $\beta \subseteq \alpha$)
- α is a superkey for R
- Each attribute A in $\beta - \alpha$ is contained in a candidate key for R .

- **Venue Table**

venue_id → venue_name, capacity, address

The candidate key is **venue_id**,

The set of key attributes are { **venue_id** }.

venue_id is a super-key for Venue Table.

Dependency is trivial.

The Venue Table is in BCNF and 3NF. No need for normalization.

- **Report Table**

report_id → details

The candidate key is **report_id**,

The set of key attributes are { **report_id** }.

report_id is a super-key for Report Table.

Dependency is trivial.

The Report Table is in BCNF and 3NF. No need for normalization.

- **Competitor Table**

competitor_id → competitor_name, competitor_age, ticket_id

The candidate key is **competitor_id**,

The set of key attributes are { **competitor_id** }.

competitor_id is a super-key for Competitor Table.

Dependency is trivial.

The Competitor Table is in BCNF and 3NF. No need for normalization.

- **Organizer Table**

organizer_id → organizer_name, competition_id

The candidate key is **organizer_id**,

The set of key attributes are { **organizer_id** }.

organizer_id is a super-key for Organizer Table.

Dependency is trivial.

The Organizer Table is in BCNF and 3NF. No need for normalization.

- **Visitor Table**

visitor_id → visitor_name, visitor_age, competition_id

The candidate key is **visitor_id**,

The set of key attributes are { **visitor_id** }.

visitor_id is a super-key for Visitor Table.

Dependency is trivial.

The Visitor Table is in BCNF and 3NF. No need for normalization.

- **Jury Table**

jury_id → jury_name, expertise, competition_id

The candidate key is **jury_id**,

The set of key attributes are { **jury_id** }.

jury_id is a super-key for Jury Table.

Dependency is trivial.

The Jury Table is in BCNF and 3NF. No need for normalization.

- **Ticket Table**

ticket_id → competition_id, competition_name, date, venue_id, venue_name

competition_id → competition_name, date

venue_id → venue_name

The candidate key is **ticket_id**,

The set of key attributes are { **ticket_id** }.

ticket_id is a super-key for Ticket Table.

Since we can get competition_name, date, venue_name from **competition_id** and

venue_id , dependency is non-trivial.This violates BCNF and 3NF.
The Ticket Table is not in BCNF and 3NF. Needs normalization.

– If we remove competition_name,date and venue_name from the table,
Dependency will become trivial and the Ticket Table will be in BCNF
and 3NF.

- **Competition Table**

competition_id→competition_name,**venue_id**,venue_name,**organizer_id**,
organizer_name,**jury_id**,jury_name,ticket_quantity,
visitor_quantity,**report_id**,date

venue_id → venue_name

organizer_id → organizer_name

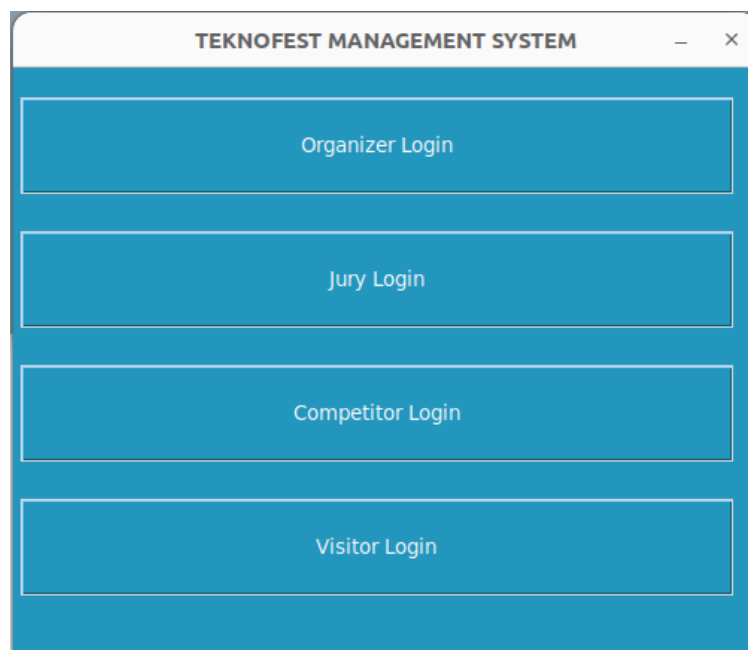
jury_id →jury_name

Since we can get venue_name, organizer_name, and jury_name from
organizer_id, venue_id and jury_id , dependency is non-trivial.This
violates BCNF and 3NF.

The Competition Table is not in BCNF and 3NF. Needs normalization.

– If we remove venue_name, organizer_name and jury_name from
the table,Dependency will become trivial and the Competition Table will be in
BCNF and 3NF.

8) QUERIES



• SELECT

- SELECT * FROM Competition

```
global cursor
clearEntries(self.entries)
cursor.execute(SELECT * FROM Competition)
```

Organizer									
Create Competition	Edit Competition	Cancel Competition	List Competitions	List Juries	List Venues	See Report	Add Report		
competition_id	competition_name	organizer_id	venue_id	ticket_quantity	visitor_quantity	jury_id	report_id	date	venue_id
1	Competition 1	1	1	150	15	1	None	2023-06-26	1
2	Competition 2	1	2	100	20	2	None	2023-07-22	2
3	Competition 3	1	3	60	10	2	None	2023-06-15	3
4	Competition 4	3	None	300	50	1	None	2023-08-26	None
5	Competition 5	2	4	20	5	3	None	2023-10-21	4
6	Competition 6	3	5	50	10	4	None	2023-07-12	5
7	Competition 7	4	5	40	15	5	None	2023-06-05	5
8	Competition 8	3	4	28	12	6	None	2023-11-09	4
9	yeni	1	1	0	1	1	None	2025-02-10	1

- SELECT * FROM Jury

```
global cursor
clearEntries(self.entries)
cursor.execute(SELECT * FROM Jury)
```

Organizer									
Create Competition	Edit Competition	Cancel Competition	List Competitions	List Juries	List Venues	See Report	Add Report		
jury_id	jury_name	competition_id	expertise	competition_name	date				
1	Ahmet	9	History	yeni	2025-02-10				
2	Mehmet	1	Math	Competition 1	2023-06-26				
3	Ayşe	5	Science	Competition 5	2023-10-21				
4	Ali	6	Human Rights	Competition 6	2023-07-12				

- SELECT * FROM Venue

Organizer									
Create Competition	Edit Competition	Cancel Competition	List Competitions	List Juries	List Venues	See Report	Add Report		
venue_id	venue_name	address	capacity						
1	Room 1	Block A, Floor 1	300						
2	Room 2	Block A, Floor 2	150						
3	Room 3	Block A, Floor 3	100						
4	Kartal	Istanbul,Kartal	200						
5	Pendik	Istanbul,Pendik	150						
6	Kartal 2	Istanbul,Kartal	20						
7	Room 7	Block B, Floor 1	60						
8	Room 8	Block B, Floor 2	90						

• INSERT

• INSERT INTO Competition (From Organizer View)

```
try:
    if not mydb.in_transaction:
        mydb.start_transaction()
    cursor.execute("""INSERT INTO Competition(competition_name,
        organizer_id,venue_id,ticket_quantity,visitor_quantity,
        jury_id,report_id,date) values(%s,%s,%s,%s,%s,%s,%s,%s)""",
        (competition_name,organizer_id,venue_id,ticket_quantity,visitor_quantity,jury_id,report_id,competition_date))
    cursor.execute("SELECT competition_id FROM Competition ORDER BY competition_id DESC LIMIT 1")
    last_id = cursor.fetchone()[0]
    cursor.execute(
        "UPDATE Jury SET competition_id=%s WHERE jury_id=%s", (last_id, jury_id))
    mydb.commit()
except mysql.connector.Error as err:
    show_warning(err.msg)
    mydb.rollback()
```

Before:

Organizer												
<div> <div>Create Competition</div> <div>Edit Competition</div> <div>Cancel Competition</div> <div>List Competitions</div> <div>List Juries</div> <div>List Venues</div> <div>See Report</div> <div>Add Report</div> </div>												
competition_id	competition_name	organizer_id	venue_id	ticket_quantity	visitor_quantity	jury_id	report_id	date	venue_id	venue_name	address	capacity
1	Competition 1	1	1	150	15	1	None	2023-06-26	1	Room 1	Block A, Floor 1	300
2	Competition 2	1	2	100	20	2	None	2023-07-22	2	Room 2	Block A, Floor 2	150
3	Competition 3	1	3	60	10	2	None	2023-06-15	3	Room 3	Block A, Floor 3	100
4	Competition 4	3	None	300	50	1	None	2023-08-26	None	None	None	None
5	Competition 5	2	4	20	5	3	None	2023-10-21	4	Kartal	Istanbul,Kartal	200
6	Competition 6	3	5	50	10	4	None	2023-07-12	5	Pendik	Istanbul,Pendik	150
7	Competition 7	4	5	40	15	5	None	2023-06-05	5	Pendik	Istanbul,Pendik	150
8	Competition 8	3	4	28	12	6	None	2023-11-09	4	Kartal	Istanbul,Kartal	200

New Item:

Create Competition

Competition Name:

yeni

Venue ID:

1

Organizer ID:

1

Jury ID:

1

Ticket Quantity:

1

Visitor Quantity:

1

Competition Date:

2025-02-10

Create Competition

After:

Organizer												
<div> <div>Create Competition</div> <div>Edit Competition</div> <div>Cancel Competition</div> <div>List Competitions</div> <div>List Juries</div> <div>List Venues</div> <div>See Report</div> <div>Add Report</div> </div>												
competition_id	competition_name	organizer_id	venue_id	ticket_quantity	visitor_quantity	jury_id	report_id	date	venue_id	venue_name	address	capacity
1	Competition 1	1	1	150	15	1	None	2023-06-26	1	Room 1	Block A, Floor 1	300
2	Competition 2	1	2	100	20	2	None	2023-07-22	2	Room 2	Block A, Floor 2	150
3	Competition 3	1	3	60	10	2	None	2023-06-15	3	Room 3	Block A, Floor 3	100
4	Competition 4	3	None	300	50	1	None	2023-08-26	None	None	None	None
5	Competition 5	2	4	20	5	3	None	2023-10-21	4	Kartal	Istanbul,Kartal	200
6	Competition 6	3	5	50	10	4	None	2023-07-12	5	Pendik	Istanbul,Pendik	150
7	Competition 7	4	5	40	15	5	None	2023-06-05	5	Pendik	Istanbul,Pendik	150
8	Competition 8	3	4	28	12	6	None	2023-11-09	4	Kartal	Istanbul,Kartal	200
9	yeni	1	1	0	1	1	None	2025-02-10	1	Room 1	Block A, Floor 1	300

- INSERT INTO Jury (From Jury View)

```
try:
    cursor.execute("""INSERT INTO Jury(jury_name,competition_id,expertise) values(%s,%s,%s)""",
                    (name, None, expertise))
except mysql.connector.Error as err:
    show_warning(err.msg)
```

Before:

Jury				
List Competitions	Attend Competition	Change Expertise	List Juries	Add Jury
jury_id	jury_name	competition_id	expertise	
1	Ahmet	9	History	
2	Mehmet	1	Math	
3	Ayşe	5	Science	
4	Ali	6	Human Rights	
5	Yakup	7	Justice	

New Item:

Add Jury

Name:

Talha

Expertise:

Economy

Add Jury

After:

Jury				
List Competitions	Attend Competition	Change Expertise	List Juries	Add Jury
jury_id	jury_name	competition_id	expertise	
1	Ahmet	9	History	
2	Mehmet	1	Math	
3	Ayşe	5	Science	
4	Ali	6	Human Rights	
5	Yakup	7	Justice	
6	Talha	8	Economy	

- INSERT INTO Competitor (From Competitor View)

```
try:
    cursor.execute("""INSERT INTO Competitor(competitor_name,age,ticket_quantity)values(%s,%s,%s)""",
                    (competitor_name,age,None))
except mysql.connector.Error as err:
    show_warning(err.msg)
```

Before:

Competitor			
List Competitions	Buy Ticket	List All Ticket	Cancel Ticket
List Competitors	Add Competitor		
competitor_id	competitor_name	age	ticket_quantity
1	Competitor 1	18	2
2	Competitor 1	22	0
3	Competitor 2	17	0
4	Competitor 3	16	0
5	Competitor 4	21	0
6	Competitor 5	28	1
7	Competitor 6	34	0
8	Competitor 7	31	0
9	Competitor 8	32	0
10	Competitor 9	5	None

New Item:

Add Competitor

Competitor Name:

yeni_yarismaci

Age:

20

Add Competitor

After:

Competitor			
List Competitions	Buy Ticket	List All Ticket	Cancel Ticket
List Competitors	Add Competitor		
competitor_id	competitor_name	age	ticket_quantity
1	Competitor 1	18	2
2	Competitor 1	22	0
3	Competitor 2	17	0
4	Competitor 3	16	0
5	Competitor 4	21	0
6	Competitor 5	28	1
7	Competitor 6	34	0
8	Competitor 7	31	0
9	Competitor 8	32	0
10	Competitor 9	5	None
11	yeni_yarismaci	20	None

• DELETE

- DELETE id FROM Competition

```
try:  
    cursor.execute("DELETE FROM Competition WHERE competition_id=%s",  
                  (competition_id,))  
except mysql.connector.Error as err:  
    show_warning(err.msg)
```

Before:

Organizer												
competition_id	competition_name	organizer_id	venue_id	ticket_quantity	visitor_quantity	jury_id	report_id	date	venue_id	venue_name	address	capacity
1	Competition 1	1	1	150	15	1	None	2023-06-26	1	Room 1	Block A, Floor 1	300
2	Competition 2	1	2	100	20	2	None	2023-07-22	2	Room 2	Block A, Floor 2	150
3	Competition 3	1	3	60	10	2	None	2023-06-15	3	Room 3	Block A, Floor 3	100
4	Competition 4	3	None	300	50	1	None	2023-08-26	None	None	None	None
5	Competition 5	2	4	20	5	3	None	2023-10-21	4	Kartal	Istanbul, Kartal	200
6	Competition 6	3	5	50	10	4	None	2023-07-12	5	Pendik	Istanbul, Pendik	150
7	Competition 7	4	5	40	15	5	None	2023-06-05	5	Pendik	Istanbul, Pendik	150
8	Competition 8	3	4	28	12	6	None	2023-11-09	4	Kartal	Istanbul, Kartal	200
9	yeni	1	1	0	1	1	None	2025-02-10	1	Room 1	Block A, Floor 1	300

Removoid Item:

Cancel Competition

Competition ID:

Cancel Competition

After:

Organizer												
competition_id	competition_name	organizer_id	venue_id	ticket_quantity	visitor_quantity	jury_id	report_id	date	venue_id	venue_name	address	capacity
1	Competition 1	1	1	150	15	1	None	2023-06-26	1	Room 1	Block A, Floor 1	300
2	Competition 2	1	2	100	20	2	None	2023-07-22	2	Room 2	Block A, Floor 2	150
3	Competition 3	1	3	60	10	2	None	2023-06-15	3	Room 3	Block A, Floor 3	100
4	Competition 4	3	None	300	50	1	None	2023-08-26	None	None	None	None
5	Competition 5	2	4	20	5	3	None	2023-10-21	4	Kartal	Istanbul, Kartal	200
6	Competition 6	3	5	50	10	4	None	2023-07-12	5	Pendik	Istanbul, Pendik	150
7	Competition 7	4	5	40	15	5	None	2023-06-05	5	Pendik	Istanbul, Pendik	150
8	Competition 8	3	4	28	12	6	None	2023-11-09	4	Kartal	Istanbul, Kartal	200

- DELETE ticket_id FROM Ticket

```
cursor.execute("DELETE FROM Ticket WHERE ticket_id= %s AND competitor_id=%s ",
              (ticket_id,competitor_id))
```

Before:

Competitor								
List Competitions	Buy Ticket	List All Ticket	Cancel Ticket	List Competitors	Add Competitor			
competition_id	competition_name	organizer_id	venue_id	ticket_quantity	jury_id	date	jury_name	expertise
1	Competition 1	1	1	150	1	2023-06-26	Ahmet	History
2	Competition 2	1	2	100	2	2023-07-22	Mehmet	Math
3	Competition 3	1	3	60	2	2023-06-15	Mehmet	Math
4	Competition 4	3	None	300	1	2023-08-26	Ahmet	History
5	Competition 5	2	4	20	3	2023-10-21	Ayşe	Science
6	Competition 6	3	5	50	4	2023-07-12	Ali	Human Rights
7	Competition 7	4	5	40	5	2023-06-05	Yakup	Justice
8	Competition 8	3	4	28	6	2023-11-09	Talha	Economy

Removid Item:

Cancel Ticket

Competitor ID:

6

Ticket ID:

1

Cancel Ticket

After:

Competitor								
List Competitions	Buy Ticket	List All Ticket	Cancel Ticket	List Competitors	Add Competitor			
competition_id	competition_name	organizer_id	venue_id	ticket_quantity	jury_id	date	jury_name	expertise
1	Competition 1	1	1	150	1	2023-06-26	Ahmet	History
2	Competition 2	1	2	100	2	2023-07-22	Mehmet	Math
3	Competition 3	1	3	60	2	2023-06-15	Mehmet	Math
4	Competition 4	3	None	300	1	2023-08-26	Ahmet	History
5	Competition 5	2	4	20	3	2023-10-21	Ayşe	Science
6	Competition 6	3	5	50	4	2023-07-12	Ali	Human Rights
7	Competition 7	4	5	40	5	2023-06-05	Yakup	Justice

- DELETE id FROM Visitor

```

try:
    cursor.execute("DELETE FROM Visitor WHERE visitor_id= %s",(visitor_id,))
except mysql.connector.Error as err:
    show_warning(err.msg)

```

Before:

Visitor			
List Competitions	Be Volunteer	List Visitor	Add Visitor
Edit Visitor	Remove Visitor		
visitor_id	visitor_name	age	competition_id
1	Visitor 1	25	None
2	Visitor 2	20	None
3	Visitor 3	30	None
4	Visitor 4	26	None
5	Visitor 5	19	None

Removid Item:

Remove Visitor

Visitor ID:

1

Remove Visitor

After:

Visitor			
List Competitions	Be Volunteer	List Visitor	Add Visitor
Edit Visitor	Remove Visitor		
visitor_id	visitor_name	age	competition_id
2	Visitor 2	20	None
3	Visitor 3	30	None
4	Visitor 4	26	None
5	Visitor 5	19	None

• UPDATE

• UPDATE Competition

```
cursor.execute("UPDATE Competition SET jury_id= %s WHERE competition_id= %s",
               (jury_id, competition_id))
```

Before:

Organizer									
Create Competition	Edit Competition	Cancel Competition	List Competitions	List Juries	List Venues	See Report	Add Report		
competition_id	competition_name	organizer_id	venue_id	ticket_quantity	visitor_quantity	jury_id	report_id	date	venue_id
1	Competition 1	1	1	150	15	1	None	2023-06-26	1
2	Competition 2	1	2	100	20	2	None	2023-07-22	2
3	Competition 3	1	3	60	10	2	None	2023-06-15	3
4	Competition 4	3	None	300	50	1	None	2023-08-26	None

Updated Item:

Edit Competition

Competition ID:

Competition Name:

Venue ID:

Organizer ID:

Jury ID:

Ticket Quantity:

Visitor Quantity:

Report ID:

Competition Date:

Update Competition

After:

Organizer									
Create Competition	Edit Competition	Cancel Competition	List Competitions	List Juries	List Venues	See Report	Add Report		
competition_id	competition_name	organizer_id	venue_id	ticket_quantity	visitor_quantity	jury_id	report_id	date	venue_id
1	edited	1	1	300	15	1	1	2025-10-02	1
2	Competition 2	1	2	100	20	2	None	2023-07-22	2
3	Competition 3	1	3	60	10	2	None	2023-06-15	3
4	Competition 4	3	None	300	50	1	None	2023-08-26	None

- UPDATE Jury

```
try:
    cursor.execute("UPDATE Jury SET expertise= %s WHERE jury_id= %s",
                    (expertise, jury_id,))
except mysql.connector.Error as err:
    show_warning(err.msg)
```

Before:

jury_id	jury_name	competition_id	expertise
1	Ahmet	9	History
2	Mehmet	1	Math
3	Ayşe	5	Science
4	Ali	6	Human Rights
5	Yakup	7	Justice
6	Talha	8	Economy

Updated Item:

Change Expertise

Jury ID:

1

Expertise:

Database

Change Expertise

After:

jury_id	jury_name	competition_id	expertise
1	Ahmet	9	Database
2	Mehmet	1	Math
3	Ayşe	5	Science
4	Ali	6	Human Rights
5	Yakup	7	Justice
6	Talha	8	Economy

- UPDATE Visitor

```

try:
    cursor.execute("UPDATE Visitor SET visitor_name = %s,age=%s WHERE visitor_id = %s",
                    (visitor_name, age, visitor_id))
except mysql.connector.Error as err:
    show_warning(err.msg)

```

Before:

Visitor			
List Competitions	Be Volunteer	List Visitor	Add Visitor
Edit Visitor	Remove Visitor		
visitor_id	visitor_name	age	competition_id
2	Visitor 2	20	None
3	Visitor 3	30	None
4	Visitor 4	26	None
5	Visitor 5	19	None

Updated Item:

Edit Visitor

Visitor ID:

2

Visitor Name:

edited

Age:

199

Edit Visitor

After:

Visitor			
List Competitions	Be Volunteer	List Visitor	Add Visitor
Edit Visitor	Remove Visitor		
visitor_id	visitor_name	age	competition_id
2	edited	199	None
3	Visitor 3	30	None
4	Visitor 4	26	None
5	Visitor 5	19	None

9) JOIN

• LEFT JOIN

```
def listCompetitions(self):
    global cursor
    clearEntries(self.entries) # Assuming this function clears existing entries

    cursor.execute("""SELECT JuryView.*, Venue.venue_name, Venue.address
    FROM JuryView
    LEFT JOIN Venue ON JuryView.venue_id = Venue.venue_id""")
```

Jury						
List Competitions	Attend Competitor	Change Expertise	List Juries	Add Jury		
competition_id	competition_name	organizer_id	venue_id	jury_id	date	venue_name address

• RIGHT JOIN

```
def listCompetitions(self):
    global cursor
    clearEntries(self.entries)
    cursor.execute("""SELECT CompetitorView.*, Jury.jury_name, Jury.expertise
    FROM CompetitorView
    RIGHT JOIN Jury ON CompetitorView.jury_id = Jury.jury_id
    WHERE CompetitorView.competition_id IS NOT NULL""")
```

Competitor								
List Competitions	Buy Ticket	List All Ticket	Cancel Ticket	List Competitors	Add Competitor			
competition_id	competition_name	organizer_id	venue_id	ticket_quantity	jury_id	date	jury_name	expertise

• FULL JOIN

MySQL doesn't support FULL JOIN. So I've union the left and right join.

```
def listCompetitions(self):
    global cursor
    clearEntries(self.entries)
    cursor.execute("""SELECT * FROM Competition LEFT JOIN Venue ON Competition.venue_id = Venue.venue_id
    UNION
    SELECT * FROM Competition RIGHT JOIN Venue ON Competition.venue_id = Venue.venue_id
    WHERE Competition.competition_id IS NOT NULL""")
```

Organizer											
Create Competition	Edit Competition	Cancel Competition	List Competitions	List Juries	List Venues	See Report	Add Report				
competition_id	competition_name	organizer_id	venue_id	ticket_quantity	visitor_quantity	jury_id	report_id	date	venue_id	venue_name	address capacity
1	1000000	1	1	1	1	1	1	2024-10-01	1	Room 1	Block A Floor 1 1000

10) TRIGGERS

• Trigger 1: Venue Capacity - Ticket Quantity

Controlling venue capacity is enough for the event's ticket quantity.

```
def addTrigger1(cursor, conn):
    cursor.execute("""CREATE TRIGGER check_capacity_trigger
    BEFORE INSERT ON Competition
    FOR EACH ROW
    BEGIN
        DECLARE venue_capacity INT;

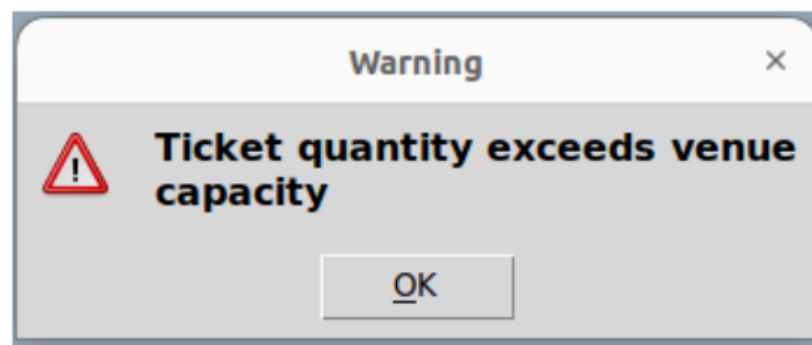
        SELECT capacity INTO venue_capacity
        FROM Venue
        WHERE venue_id = NEW.venue_id;

        IF NEW.ticket_quantity > venue_capacity THEN
            SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Ticket quantity exceeds venue capacity';
        END IF;
    END;""")
    conn.commit()
```

venue_id	venue_name	address	capacity
1	Room 1	Block A, Floor 1	300
2	Room 2	Block A, Floor 2	150

Form fields for creating a competition:

- Competition Name: robotaksi
- Venue ID: 2
- Organizer ID: 1
- Jury ID: 1
- Ticket Quantity: 500
- Visitor Quantity: 200
- Competition Date: 2025-10-02
- Create Competition button



• Trigger 2: New Competition Date

New competition date must be in the future.

```
def addTrigger2(cursor, conn):
    cursor.execute("""CREATE TRIGGER check_future_date_trigger
    BEFORE INSERT ON Competition
    FOR EACH ROW
    BEGIN
        IF NEW.date <= CURDATE() THEN
            SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Competition date must be in the future';
        END IF;
    END;""")
    conn.commit()
```

Competition Name:
cse414

Venue ID:
1

Organizer ID:
1

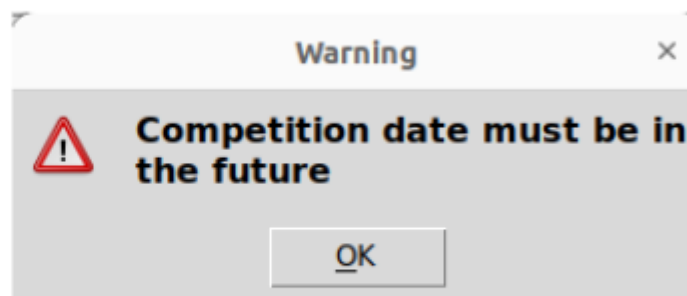
Jury ID:
1

Ticket Quantity:
10

Visitor Quantity:
5

Competition Date:
2024-06-10

Create Competition



When I tried to add an event in past date, it doesn't allow.

• Trigger 3: Near Future Competition

Competitions closer than 1 month cannot be canceled.

```
def addTrigger3(cursor, conn):
    cursor.execute("""CREATE TRIGGER check_month_until_competition_delete_trigger
    BEFORE DELETE ON Competition
    FOR EACH ROW
    BEGIN
        IF OLD.date <= DATE_ADD(CURDATE(), INTERVAL 1 MONTH) THEN
            SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Competition must be at least one month away';
        END IF;
    END;""")
    conn.commit()
```

Organizer

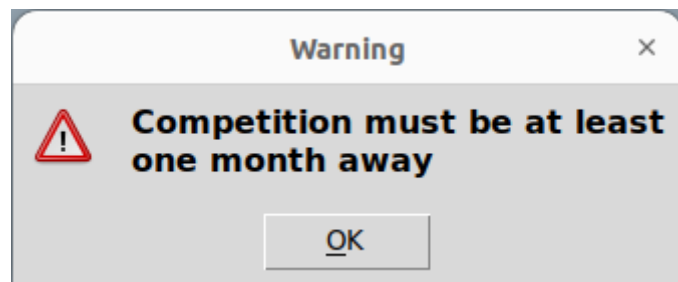
Create Competition Edit Competition Cancel Competition List Competitions List Juries List Venues See Report Add Report

competition_id	competition_name	organizer_id	venue_id	ticket_quantity	visitor_quantity	jury_id	report_id	date	venue_id	
1	edited	1	1	300	15	1	1	2025-10-02	1	R
2	Competition 2	1	2	100	20	2	None	2023-07-22	2	P
3	Competition 3	1	3	60	10	2	None	2023-06-15	3	P
4	Competition 4	3	None	300	50	1	None	2023-08-26	None	N
5	Competition 5	2	4	20	5	3	None	2023-10-21	4	K
6	Competition 6	3	5	50	10	4	None	2023-07-12	5	P
7	Competition 7	4	5	40	15	5	None	2023-06-05	5	P
8	Competition 8	3	4	28	12	6	None	2023-11-09	4	K
9	yeni	1	1	0	1	1	None	2025-02-10	1	P

Cancel Competition

Competition ID:
6

Cancel Competition



• Trigger 4: Ticket Quantity

Can not buy a ticket if there is no ticket.

```
def addTrigger4(cursor, conn):
    cursor.execute("""CREATE TRIGGER check_ticket_quantity_trigger
    BEFORE UPDATE ON Competition
    FOR EACH ROW
    BEGIN
        IF NEW.ticket_quantity < 0 THEN
            SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'No Ticket';
        END IF;
    END;""")
    conn.commit()
```

Competitor

List Competitions Buy Ticket List All Ticket Cancel Ticket List Competitors Add Competitor

competition_id	competition_name	organizer_id	venue_id	ticket_quantity	jury_id	date	jury_name	expertise
1	edited	1	1	300	1	2025-10-02	Ahmet	Database
2	Competition 2	1	2	100	2	2023-07-22	Mehmet	Math
3	Competition 3	1	3	60	2	2023-06-15	Mehmet	Math
4	Competition 4	3	None	300	1	2023-08-26	Ahmet	Database
5	Competition 5	2	4	20	3	2023-10-21	Ayşe	Science
6	Competition 6	3	5	50	4	2023-07-12	Ali	Human Rights
7	Competition 7	4	5	40	5	2023-06-05	Yakup	Justice
8	Competition 8	3	4	28	6	2023-11-09	Talha	Economy
9	yeni	1	1	0	1	2025-02-10	Ahmet	Database


Buy Ticket

Competitor ID: 1

Competition ID: 9

Buy Ticket

Warning

 **No Ticket**

OK

Buy Ticket

Competitor ID: 1

Competition ID: 9

Buy Ticket

• Trigger 5: Visitor Age

Visitors must be over 18 years of age to be volunteer in the competition.

```
def addTrigger5(cursor, conn):
    cursor.execute("""CREATE TRIGGER check_visitor_age_trigger
    BEFORE UPDATE ON Visitor
    FOR EACH ROW
    BEGIN
        IF OLD.age < 18 THEN
            SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Visitor age must be at least 18.';
        END IF;
    END;""")
    conn.commit()
```


Visitor

List Competitions
Be Volunteer
List Visitor
Add Visitor
Edit Visitor
Remove Visitor

visitor_id	visitor_name	age	competition_id
2	Visitor 2	15	None
3	Visitor 3	30	None
4	Visitor 4	26	None
5	Visitor 5	19	None

Volunteer ✕
 Visitor ID:

 Competition ID:

Volunteer

Warning ✕

Visitor age must be at least 18.

OK

12) VIEWS

• View 1: Competitor View

Competitors can't see report_id and visitor_quantity.

```
# Competitors can't see report_id and visitor_quantity.
def createView1(cursor, conn):
    cursor.execute("""CREATE VIEW CompetitorView AS
        SELECT competition_id, competition_name, organizer_id, venue_id, ticket_quantity, jury_id, date
        FROM Competition;""")
    conn.commit()
```

Competitor								
List Competitions	Buy Ticket	List All Ticket	Cancel Ticket	List Competitors	Add Competitor			
competition_id	competition_name	organizer_id	venue_id	ticket_quantity	jury_id	date	jury_name	expertise
1	edited	1	1	300	1	2025-10-02	Ahmet	Database
2	Competition 2	1	2	100	2	2023-07-22	Mehmet	Math
3	Competition 3	1	3	60	2	2023-06-15	Mehmet	Math
4	Competition 4	3	None	300	1	2023-08-26	Ahmet	Database
5	Competition 5	2	4	20	3	2023-10-21	Ayşe	Science

• View 2: Visitor View

Visitors can't see report_id.

```
# Visitor can't see report_id.
def createView2(cursor, conn):
    cursor.execute("""CREATE VIEW VisitorView AS
    SELECT competition_name,organizer_id,venue_id,ticket_quantity,visitor_quantity,jury_id,date
    FROM Competition;""")
    conn.commit()
```

Visitor						
List Competitions	Be Volunteer	List Visitor	Add Visitor	Edit Visitor	Remove Visitor	
competition_name	organizer_id	venue_id	ticket_quantity	visitor_quantity	jury_id	date
edited	1	1	300	15	1	2025-10-02
Competition 2	1	2	100	20	2	2023-07-22
Competition 3	1	3	60	10	2	2023-06-15
Competition 4	3	None	300	50	1	2023-08-26
Competition 5	2	4	20	5	3	2023-10-21
Competition 6	3	5	50	10	4	2023-07-12
Competition 7	4	5	40	15	5	2023-06-05
Competition 8	3	4	28	12	6	2023-11-09
yeni	1	1	0	1	1	2025-02-10

• View 3: Jury View

Juries can't see competitions that have juries.

```
# Juries can't see competitions that have juries.
def createView3(cursor, conn):
    cursor.execute("""CREATE VIEW JuryView AS
    SELECT competition_id,competition_name,organizer_id,venue_id,jury_id,date
    FROM Competition
    WHERE jury_id IS NULL;""")
    conn.commit()
```

Jury							
List Competitions	Attend Competition	Change Expertise	List Juries	Add Jury			
competition_id	competition_name	organizer_id	venue_id	jury_id	date	venue_name	address

• View 4: Jury Info View

Juries can see other juries' events

```
# Juries can see other juries competitions
def createView4(cursor, conn):
    cursor.execute("""CREATE VIEW JuryInfo AS
    SELECT Jury.*,Competition.competition_name,Competition.date
    FROM Jury,Competition
    WHERE Jury.competition_id = Competition.competition_id;""")
    conn.commit()
```

Jury					
List Competitions	Attend Competition	Change Expertise	List Juries	Add Jury	
jury_id	jury_name	competition_id	expertise		
1	Ahmet	9	Database		
2	Mehmet	1	Math		
3	Ayşe	5	Science		
4	Ali	6	Human Rights		

• View 5: Ticket View

Competitors can see jury's name and topic

```
# Competitors can see juries name and expertise
def createView5(cursor, conn):
    cursor.execute("""CREATE VIEW TicketView AS
        SELECT Ticket.*,Jury.jury_name,Jury.expertise
        FROM Ticket,Jury
        WHERE Jury.competition_id = Ticket.competition_id;""")
    conn.commit()
```

Competitor								
List Competitions	Buy Ticket	List All Ticket	Cancel Ticket	List Competitors	Add Competitor			
ticket_id	competition_id	competition_name	competition_date	venue_id	venue_name	competitor_id	jury_name	expertise
6	9	yeni	2025-02-10	1	Room 1	6	Ahmet	Database
8	8	Competition 8	2023-11-09	4	Kartal	1	Talha	Economy

13) TRANSACTIONS

• Transaction 1: Volunteering

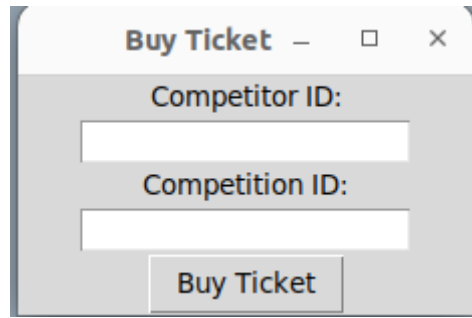
```
def be_volunteer(self):
    global cursor, mydb
    visitor_id = self.visitor_id_entry.get()
    competition_id = self.competition_id_entry.get()

    try:
        if not mydb.in_transaction:
            mydb.start_transaction()
            cursor.execute("UPDATE Visitor SET competition_id=%s WHERE visitor_id= %s", (competition_id,visitor_id))
            cursor.execute("UPDATE Competition SET visitor_quantity=visitor_quantity-1 WHERE competition_id= %s", (competition_id,))
            mydb.commit()
    except mysql.connector.Error as err:
        show_warning(err.msg)
        mydb.rollback()

    self.root.destroy()
```

• Transaction 2: Buying Ticket

```
try:
    if not mydb.in_transaction:
        mydb.start_transaction()
    cursor.execute("SELECT competition_name, date, venue_id FROM Competition WHERE competition_id= %s",(competition_id,))
    competition_details = cursor.fetchone()
    if competition_details is not None:
        competition_name, competition_date, venue_id = competition_details
        cursor.execute("SELECT venue_name FROM Venue WHERE venue_id= %s", (venue_id,))
        venue_name = cursor.fetchone()
        cursor.execute("""INSERT INTO Ticket(competition_id,competition_name,competition_date,
            venue_id,venue_name,competitor_id) values(%s,%s,%s,%s,%s,%s)""",
            (competition_id,competition_name,competition_date,venue_id,venue_name[0],competitor_id))
        cursor.execute("UPDATE Competition SET ticket_quantity = ticket_quantity -1 WHERE competition_id=%s",
            (competition_id,))
        cursor.execute("UPDATE Competitor SET ticket_quantity = ticket_quantity +1 WHERE competitor_id=%s",
            (competitor_id,))
        mydb.commit()
except mysql.connector.Error as err:
    show_warning(err.msg)
    mydb.rollback()
```



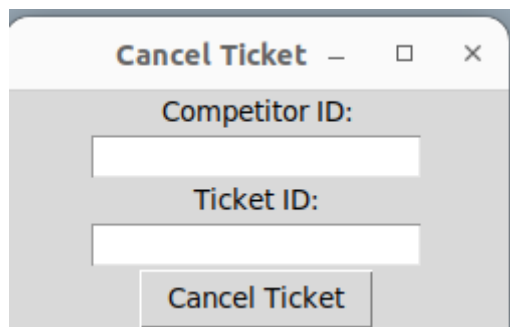
A dialog box titled "Buy Ticket" with a close button. It contains two input fields: "Competitor ID:" and "Competition ID:". Below the fields is a button labeled "Buy Ticket".

id	competition_id	competitor_id	venue_id	ticket_quantity	visitor_quantity	competition_date	competitor_name	ticket_type
8	Competition 8	3	4	28	6	2023-11-09	Talha	Economy

After buying ticket, ticket quantity decreases.

• Transaction 3: Canceling Ticket

```
try:
    if not mydb.in_transaction:
        mydb.start_transaction()
    cursor.execute("SELECT competition_id FROM Ticket WHERE ticket_id= %s", (ticket_id,))
    competition_id = cursor.fetchone()
    cursor.execute("UPDATE EVENT SET ticket_quantity= ticket_quantity+1 WHERE competition_id= %s",
                  (competition_id[0],))
    cursor.execute("DELETE FROM Ticket WHERE ticket_id= %s AND competitor_id=%s ",
                  (ticket_id, competitor_id))
    cursor.execute("UPDATE Competitor SET ticket_quantity = ticket_quantity -1 WHERE competitor_id=%s",
                  (competitor_id,))
    mydb.commit()
except mysql.connector.Error as err:
    mydb.rollback()
    show_warning(err.msg)
```



A dialog box titled "Cancel Ticket" with a close button. It contains two input fields: "Competitor ID:" and "Ticket ID:". Below the fields is a button labeled "Cancel Ticket".

id	competition_id	competitor_id	venue_id	ticket_quantity	visitor_quantity	competition_date	competitor_name	ticket_type
8	Competition 8	3	4	28	6	2023-11-09	Talha	Economy

After canceling ticket, ticket quantity increases.

• Transaction 4: Attending Competition - Jury

```
try:
    if not mydb.in_transaction:
        mydb.start_transaction()
    cursor.execute("""INSERT INTO Competition(competition_name,
        organizer_id,venue_id,ticket_quantity,visitor_quantity,
        jury_id,report_id,date) values(%s,%s,%s,%s,%s,%s,%s,%s)""",
                  (competition_name,organizer_id,venue_id,ticket_quantity,visitor_quantity,jury_id,None,competition_date))
    cursor.execute("SELECT competition_id FROM Competition ORDER BY competition_id DESC LIMIT 1")
    last_id = cursor.fetchone()[0]
    cursor.execute(
        "UPDATE Jury SET competition_id=%s WHERE jury_id=%s", (last_id, jury_id))
    mydb.commit()
except mysql.connector.Error as err:
    show_warning(err.msg)
    mydb.rollback()
```

14) PROCEDURES

• Procedure 1: AddReport

```
def createProcedure1(cursor,conn):
    cursor.execute("""CREATE PROCEDURE AddReport(IN Info VARCHAR(200))
    BEGIN
        INSERT INTO Report(details) values(Info);
    END;""")
    conn.commit()
```

```
def addReport(cursor, conn):
    cursor.callproc("AddReport", ["PDR report template"])
    cursor.callproc("AddReport", ["first report template"])
    cursor.callproc("AddReport", ["second report template"])
    cursor.callproc("AddReport", ["final report template"])
    cursor.callproc("AddReport", ["demo report template"])
    conn.commit()
```

• Procedure 2: AddVenue

```
def createProcedure2(cursor, conn):
    cursor.execute("""CREATE PROCEDURE AddVenue(IN name VARCHAR(50),IN addr VARCHAR(100),IN cap INT)
    BEGIN
        INSERT INTO Venue(venue_name,address,capacity) values(name,addr,cap);
    END;""")
    conn.commit()
```

```
def addVenue(cursor, conn):
    cursor.callproc("AddVenue",["Room 1","Block A, Floor 1",300])
    cursor.callproc("AddVenue",["Room 2", "Block A, Floor 2", 150])
    cursor.callproc("AddVenue",["Room 3", "Block A, Floor 3", 100])
    cursor.callproc("AddVenue",["Kartal", "Istanbul,Kartal", 200])
    cursor.callproc("AddVenue",["Pendik", "Istanbul,Pendik", 150])
    cursor.callproc("AddVenue",["Kartal 2", "Istanbul,Kartal", 20])
    cursor.callproc("AddVenue",["Room 7", "Block B, Floor 1", 60])
    cursor.callproc("AddVenue",["Room 8", "Block B, Floor 2", 90])
    conn.commit()
```

• Procedure 3: AddOrganizer

```
def createProcedure3(cursor,conn):
    cursor.execute("""CREATE PROCEDURE AddOrganizer(IN name VARCHAR(50),IN id INT)
    BEGIN
        INSERT INTO Organizer(organizer_name,competition_id) values(name,id);
    END;""")
    conn.commit()
```

```
def addJury(cursor, conn):
    cursor.execute("INSERT INTO Jury(jury_name,competition_id,expertise) values(%s,%s,%s)",
    ("Ahmet",2, "History"))
    cursor.execute("INSERT INTO Jury(jury_name,competition_id,expertise) values(%s,%s,%s)",
    ("Mehmet",1,"Math"))
    cursor.execute("INSERT INTO Jury(jury_name,competition_id,expertise) values(%s,%s,%s)",
    ("Ayşe", 5, "Science"))
    cursor.execute("INSERT INTO Jury(jury_name,competition_id,expertise) values(%s,%s,%s)",
    ("Ali", 6, "Human Rights"))
    cursor.execute("INSERT INTO Jury(jury_name,competition_id,expertise) values(%s,%s,%s)",
    ("Yakup", 7, "Justice"))
    cursor.execute("INSERT INTO Jury(jury_name,competition_id,expertise) values(%s,%s,%s)",
    ("Talha", 8, "Economy"))
    conn.commit()
```