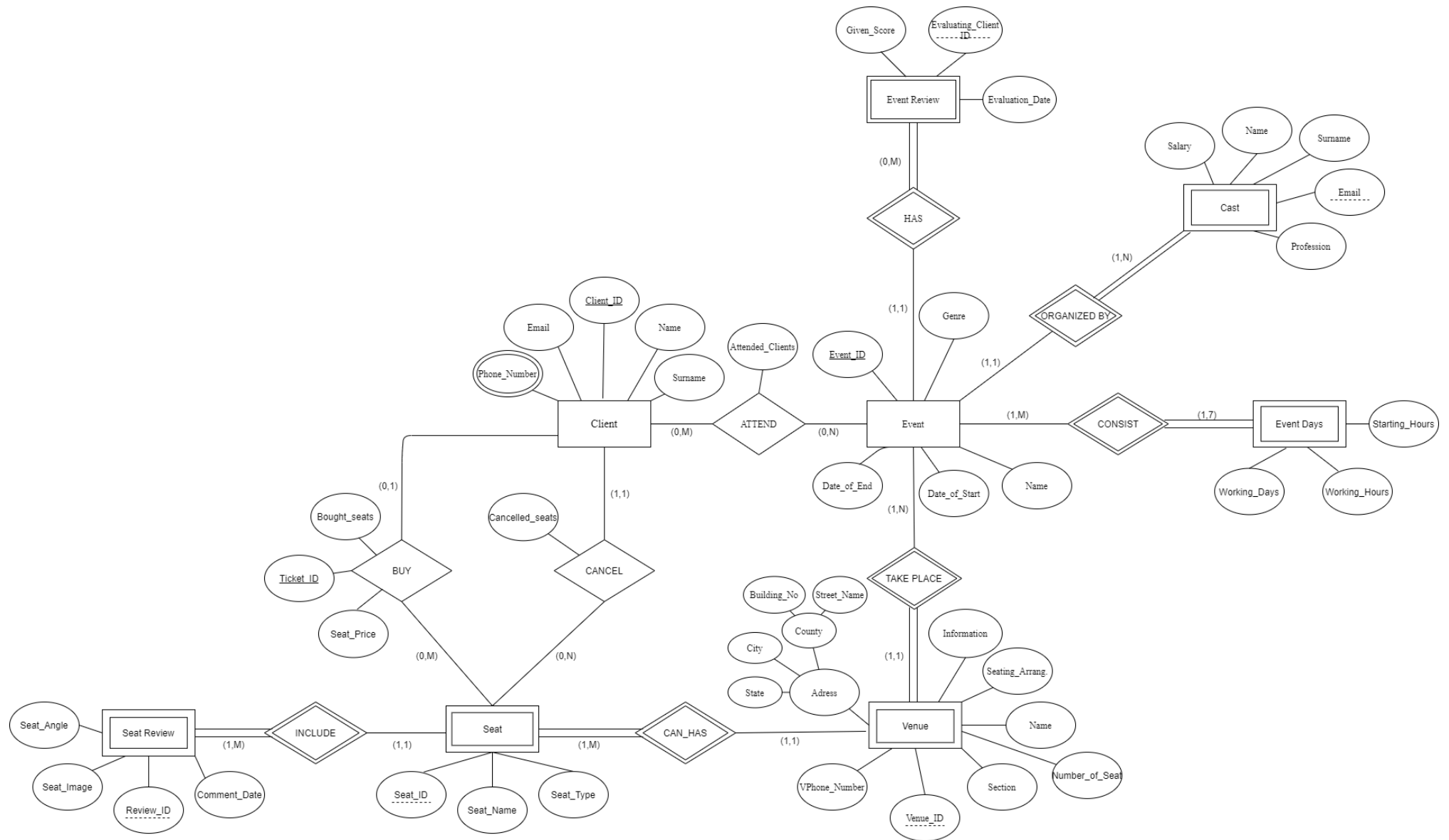


The requirements for the online ticketing system database

- The ticketing system based on the events. Each event has a unique ID, a unique name, genre, beginning date and ending date.
- An event has several (including 0) event review and each event review must associated with an event. When the client enters the event review, he/she can see the given score, can see evaluation date of the event, and he/she can evaluate the event himself/herself. In order to evaluate the event, client must enter client's id.
- An event organized by at least one cast and each cast must organize at most one event. Each cast has unique email, name surname, profession and salary.
- Events consist event days. An event must take place on some day(s) in a week. Since event days dependent on events, it should match at least one event. When the client attends the event, he/she should be able to see which days and time intervals available of that event.
- Each event takes place in just one venue and each venue can host more than one events. Each venue has unique venue id, name section, phone number, seating arrangement, information about the event, and address. In addition, one venue can has at least one seat but seat can only belong to one venue.
- Clients can attend several events and system keeps track of the clients who joined the event. Each client has a unique ID, name, surname, email and clients may have several phone numbers.
- The client can buy seat(s) with paying seat price and each seat known by its type (VIP, standard etc.), name (A1,B1 etc.) and its unique id. When client bought the seat(s), system keep track of bought seat(s) and records the purchased ticket with its id and price. In addition, client can cancel his/her seat or seats and seat(s) is cancelled by purchaser only. The system also keep track of cancelled seat(s).
- Every seat include seat review or reviews. This mean that client can see seats' angle, image, and upload date (comment date) of review if he/she want. Moreover, each seat review has a unique review id.

The ER conceptual schema diagram for the **ONLINE TICKETING SYSTEM** database



Reducing ER diagram to relational schema

Step1.mapping of strong entity sets

Client

<u>Client_ID</u>	Name	Surname	Email
------------------	------	---------	-------

Client_PhoneNumber

Client_ID	Phone_Number
-----------	--------------

Event

<u>Event_ID</u>	Name	Genre	Date_of_Start	Date_of_End
-----------------	------	-------	---------------	-------------

Step2.mapping a weak entity sets

Event Review

<u>Event_ID</u>	<u>Evaluating_Client_ID</u>	Evaluation_Date	Given_Score
-----------------	-----------------------------	-----------------	-------------

Cast

<u>Event_ID</u>	<u>Email</u>	Name	Surname	Profession	Salary
-----------------	--------------	------	---------	------------	--------

Event Days

<u>Event_ID</u>	Working_Days	Starting_Hours	Working_Hours
-----------------	--------------	----------------	---------------

Venue

<u>Event_ID</u>	<u>Venue_ID</u>	VPhone_Number	Name	Info.	Seating_Arr.	Section	State	City	County	Street_Name	Building_#	#_of_Seat
-----------------	-----------------	---------------	------	-------	--------------	---------	-------	------	--------	-------------	------------	-----------

Seat

<u>Client_ID</u>	<u>Event_ID</u>	<u>Venue_ID</u>	<u>Seat_ID</u>	Seat_Name	Seat_Type
------------------	-----------------	-----------------	----------------	-----------	-----------

Seat Review

<u>Client_ID</u>	<u>Event_ID</u>	<u>Venue_ID</u>	<u>Seat_ID</u>	<u>Review_ID</u>	Seat_Angle	Seat_Image	Comment_Date
------------------	-----------------	-----------------	----------------	------------------	------------	------------	--------------

Step3.mapping of relationship sets

Attend	<u>Client_ID</u>	<u>Event_ID</u>	Attended_Clients
	→Client	→Event	

Buy	<u>Seat_ID</u>	<u>Ticket_ID</u>	Seat_Price	Bought_seats
	→Seat			

Cancel	<u>Seat_ID</u>	Cancelled_seats
	→Seat	

Step4.combination of schemas

Client	<u>Client_ID</u>	Name	Surname	Email
--------	------------------	------	---------	-------

Client_PhoneNumber	<u>Client_ID</u>	Phone_Number
	→Client	

Event	<u>Event_ID</u>	Name	Genre	Date_of_Start	Date_of_End
-------	-----------------	------	-------	---------------	-------------

Cast	<u>Event_ID</u>	<u>Email</u>	Name	Surname	Profession	Salary
------	-----------------	--------------	------	---------	------------	--------

→Event

Event Review	<u>Event_ID</u>	<u>Evaluating_Client_ID</u>	Evaluation_Date	Given_Score
--------------	-----------------	-----------------------------	-----------------	-------------

→Event

Venue	<u>Event_ID</u>	<u>Venue_ID</u>	VPhone_Num	Name	Info.	Seating_Arr.	Section	State	City	County	Str._Name	Building_#	#_of_Seat
-------	-----------------	-----------------	------------	------	-------	--------------	---------	-------	------	--------	-----------	------------	-----------

→Event

Seat	<u>Client_ID</u>	<u>Event_ID</u>	<u>Venue_ID</u>	<u>Seat_ID</u>	Seat_Name	Seat_Type
------	------------------	-----------------	-----------------	----------------	-----------	-----------

→Client

→Event

→Venue

Seat Review	<u>Client_ID</u>	<u>Event_ID</u>	<u>Venue_ID</u>	<u>Seat_ID</u>	<u>Review_ID</u>	Seat_Angle	Seat_Image	Comment_Date
-------------	------------------	-----------------	-----------------	----------------	------------------	------------	------------	--------------

→Client

→Event

→Venue

→Seat

Event Days	<u>Event_ID</u>	Working_Days	Starting_Hours	Working_Hours
------------	-----------------	--------------	----------------	---------------

→Event

Attend	<u>Client_ID</u>	<u>Event_ID</u>	Attended_Clients
--------	------------------	-----------------	------------------

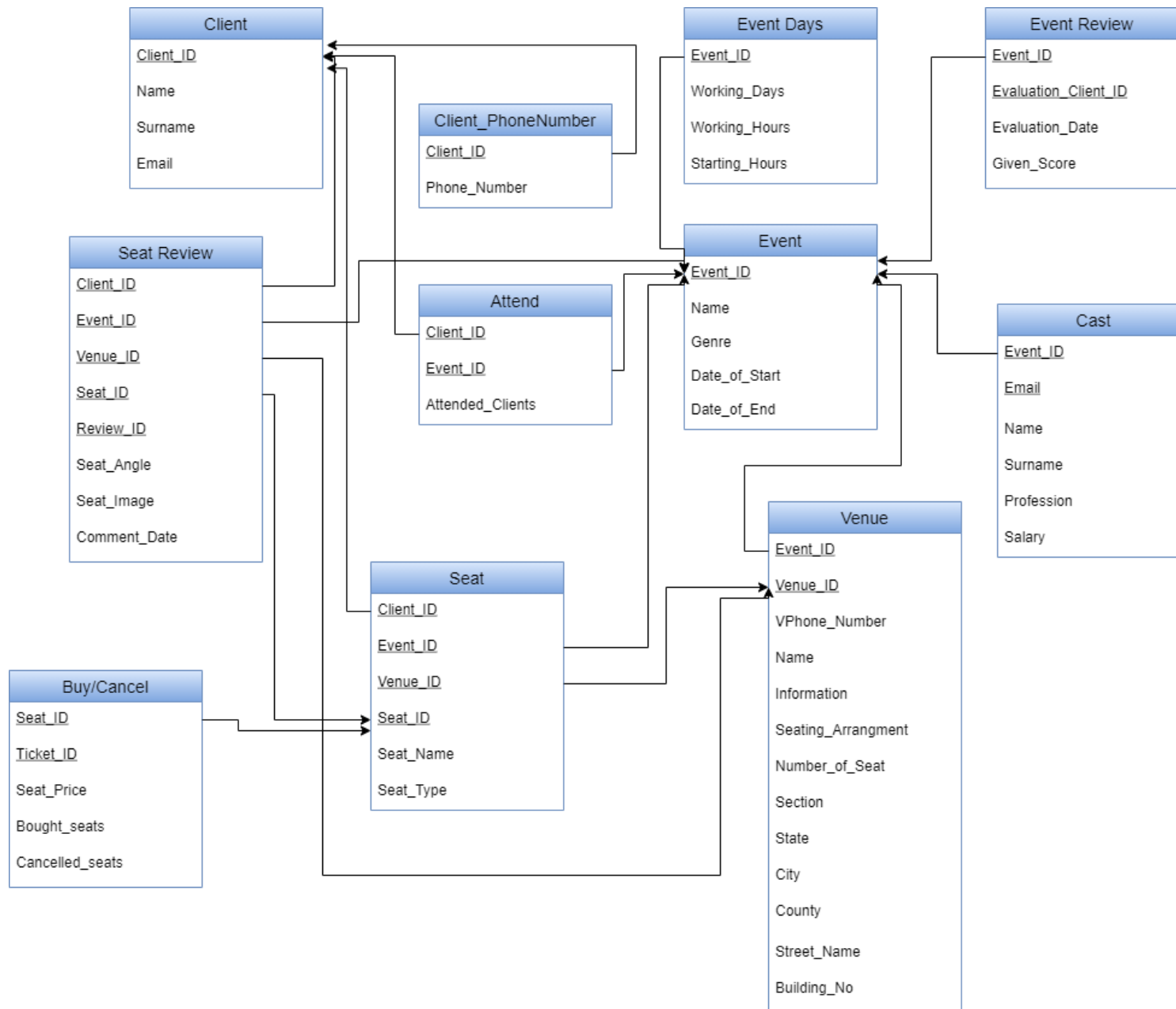
→Client

→Event

Buy/Cancel	<u>Seat_ID</u>	<u>Ticket_ID</u>	Seat_Price	Bought_seats	Cancelled_seats
------------	----------------	------------------	------------	--------------	-----------------

→Seat

The schema diagram for **online ticketing system** database



SQL SCRIPTS of Project

Client Table

```
create table Client (  
    Client_ID char(11) not null,  
    Name varchar(45) not null,  
    Surname varchar(45) not null,  
    primary key (Client_ID));
```

Client_PhoneNumber Table

```
create table Client_PhoneNumber (  
    Client_ID char(11) not null,  
    Phone_Number varchar(19) not null,  
    primary key (Client_ID, Phone_Number),  
    foreign key (Client_ID) references Client  
    on delete cascade  
    on update cascade);
```

Event Table

```
create table Event (  
    Event_ID varchar(11) not null,  
    Name varchar(45) not null,  
    Genre varchar(45) not null,  
    Date_of_Start date not null,  
    Date_of_End date not null,  
    primary key (Event_ID));
```

Cast Table

```
create table Castt (  
    CastEventID varchar(11) not null,  
    Name varchar(45) not null,  
    Surname varchar(45) not null,  
    Profession varchar(45) not null,  
    Email varchar(100) not null,  
    Salary varchar(7),  
    primary key (CastEventID, Email),  
    foreign key(CastEventID) references event  
    on delete cascade  
    on update cascade);
```

Event Review Table

```
create table Event_Review (  
    ReviewEventID varchar(11) not null,  
    Evaluating_Client_ID char(11) not null,  
    Evaluating_Date date not null,  
    Given_Score numeric(3,1) not null,  
    primary key (ReviewEventID, Evaluating_Client_ID),  
    foreign key(ReviewEventID) references Event  
    on delete cascade  
    on update cascade);
```

Venue Table

```
create table Venue (  
    VenueEventID varchar(11) not null,  
    Venue_ID varchar(11) not null,  
    VPhone_Number char(10) not null,  
    Name varchar(100) not null,  
    Section varchar(20) not null,  
    Information varchar(200) not null,  
    Seating_Arrangement bytea,  
    Number_of_Seat int not null,  
    State varchar(100) not null,  
    City varchar(163) not null,  
    County varchar(163) not null,  
    Street_Name varchar(163) not null,  
    Building_No varchar(10) not null,  
    unique (Venue_ID),  
    primary key(VenueEventID, Venue_ID),  
    foreign key(VenueEventID) references Event  
    on delete cascade  
    on update cascade);
```

Seat Table

```
create table Seat (  
    Client_ID char(11) not null,  
    Event_ID varchar(11) not null,  
    Venue_ID varchar(11) not null,  
    Seat_ID varchar(11) not null,  
    Seat_Name varchar(5) not null,  
    Seat_Type varchar(8) not null,
```



```

unique (Seat_ID),
primary key (Client_ID, Event_ID, Venue_ID, Seat_ID),
foreign key(Client_ID) references Client
on delete cascade
on update cascade,
foreign key(Event_ID) references Event
on delete cascade
on update cascade,
foreign key(Venue_ID) references Venue(Venue_ID)
on delete cascade
on update cascade);

```

Seat Review Table

```

create table Seat_Review (
    Client_ID char(11) not null,
    Event_ID varchar(11) not null,
    Venue_ID varchar(11) not null,
    Seat_ID char(11) not null,
    Review_ID varchar(11) not null,
    Seat_Angle numeric(3,1) not null,
    Seat_Image bytea,
    Comment_Date date not null,
    primary key (Client_ID, Event_ID, Venue_ID, Seat_ID),
    foreign key(Client_ID) references Client
    on delete cascade
    on update cascade,
    foreign key(Event_ID) references Event
    on delete cascade
    on update cascade,
    foreign key(Venue_ID) references Venue(Venue_ID)
    on delete cascade
    on update cascade,
    foreign key(Seat_ID) references Seat(Seat_ID)
    on delete cascade
    on update cascade);

```

Event Days Table

```

create table Event_days(
    Event_ID varchar(11) not null,
    Working_Days varchar(50) not null,
    Starting_Hours time not null,
    Working_Hours varchar(5) not null,

```

```
primary key(Event_ID),
foreign key(Event_ID) references Event
on delete cascade
on update cascade);
```

Attend Table

```
create table Attend(
  Client_ID char(11) not null,
  Event_ID varchar(11) not null,
  Attended_Clients boolean not null,
  primary key (Client_ID ,Event_ID),
  foreign key(Client_ID) references Client
  on delete cascade
  on update cascade,
  foreign key(Event_ID) references Event
  on delete cascade
  on update cascade);
```

Buy/Cancel Table

```
create table Buy_Cancel(
  Seat_ID varchar(11) not null,
  Ticket_ID varchar(11) not null,
  Seat_Price numeric(6,2)not null,
  Bought_seats boolean not null,
  Cancelled_seats boolean not null,
  primary key (Seat_ID),
  foreign key(Seat_ID) references Seat(Seat_ID)
  on delete cascade
  on update cascade);
```

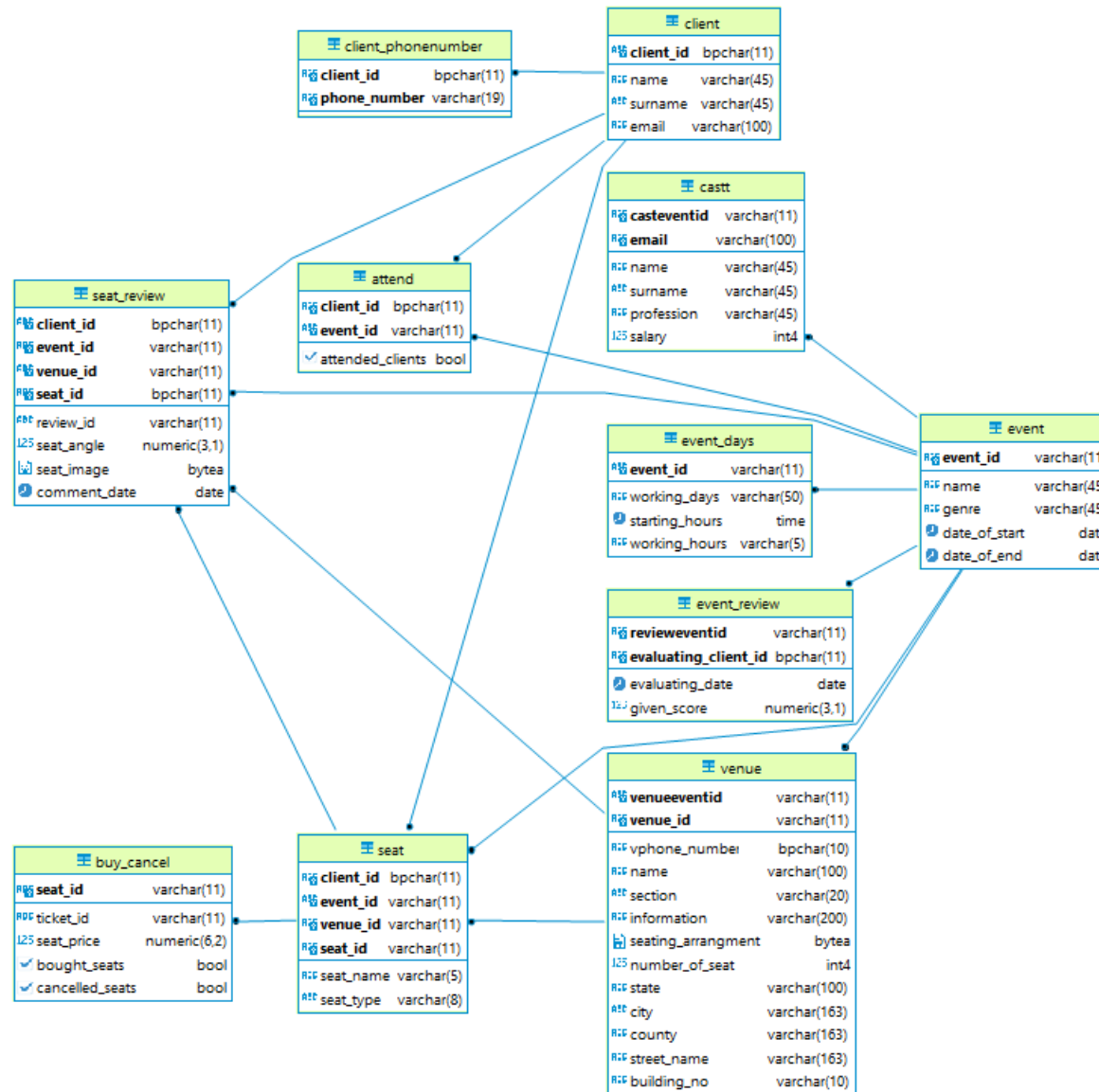
Cast View Table

```
create view Cast_View as
  select CastEventID,Name,Surname,Profession,Email
  from Castt;
```

Client Alter Table

```
alter table Client
  add column Email varchar(100);
```

The diagram that is generated on DBBeaver



*All diagrams can be accessed via this link: <https://drive.google.com/file/d/1PdIK26y0mG3gX6hKflqOtlFs851hI918/view?usp=sharing>

Project Phase #2

Queries: createdb.sql

1.Insertation

Three sample statements of insertation

Insert 1: Insert a new client to Client Table.

insert into Client

values('18098680134', 'Günsu', 'Günaydın');

Insert 2: Insert a new event to Event Table.

insert into Event

values('1111', '12 Angry Men', 'Cinema', '2020-05-16', '2020-05-17');

Insert 3: Insert a new cast to Cast View Table.

insert into Cast_View

values('1112', 'Adile', 'Naşit', 'Actor', 'adile.nasit@gmail.com');

2. Updating and Deletion

Update 1: Update existing client in Client Table.

update Client

set Surname = 'Koçyiğit'

where Surname = 'Günaydın';

Before updating Client Table

	client_id	name	surname
1	18098680134	Günsu	Günaydın
2	16429085948	Fuat Yiğit	Koçyiğit
3	14945614554	Mehmet Batuhan	Şenol
4	11985981251	Umut	Fırat
5	18958959898	Ege	Aslan
6	15165195818	Berra	Çam
7	18951951581	Yeşim	Mete
8	19848948981	Asena	Anık
9	19846516516	Hakan	Gürgün
10	11651526516	Miray	Erdemli
11	15995095950	Başak	Arslan
12	15020955900	Berkan	Güneş
13	10020566506	Elif	Işık
14	10561898914	Esra	Koçak

After updating Client Table

	client_id	name	surname
23	14845919514	Avşar	Coşkun
24	18915615959	Kemalcan	Aydoğdu
25	16518948912	Hasan	Ünal
26	18561591599	Göksu	Tümer
27	11591985191	Umay	Güven
28	12059591958	Tuna	Özdemir
29	18956159198	Albina	Gülerman
30	17988948919	Asya	Aktaş
31	10259059005	Öykü	Davud
32	13473735750	Lisa	Manoban
33	13068841891	Jisoo	Kim
34	16841818154	Henry	Federick
35	16841818177	Caleb	Stone
36	18098680134	Günsu	Koçyiğit

Delete 1: When an **client** is deleted, referenced rows of other tables are also deleted automatically due to cascading effects of Client_ID.

delete from Client

where Client_ID = '16429085948';

Before deletion on Client Table

	client_id	name	surname
1	16429085948	Fuat Yiğit	Koçyiğit
2	14945614554	Mehmet Batuhan	Şenol
3	11985981251	Umut	Fırat
4	18958959898	Ege	Aslan
5	15165195818	Berra	Çam
6	18951951581	Yeşim	Mete
7	19848948981	Asena	Anık
8	19846516516	Hakan	Gürgün
9	11651526516	Miray	Erdemli
10	15995095950	Başak	Arslan
11	15020955900	Berkan	Güneş
12	10020566506	Elif	Işık
13	10561898914	Esra	Koçak
14	19859819785	Meryem	Demircan
15	15814985519	Neslihan	Erdoğan

After deletion on Client Table

	client_id	name	surname
1	14945614554	Mehmet Batuhan	Şenol
2	11985981251	Umut	Fırat
3	18958959898	Ege	Aslan
4	15165195818	Berra	Çam
5	18951951581	Yeşim	Mete
6	19848948981	Asena	Anık
7	19846516516	Hakan	Gürgün
8	11651526516	Miray	Erdemli
9	15995095950	Başak	Arslan
10	15020955900	Berkan	Güneş
11	10020566506	Elif	Işık
12	10561898914	Esra	Koçak
13	19859819785	Meryem	Demircan
14	15814985519	Neslihan	Erdoğan
15	15620505060	Pınar	Ertenoğlu

Before deletion on Client Table (Seat Table)

	client_id	event_id	venue_id	seat_id	seat_name	seat_type
1	18098680134	1214	1	000001	B1	Standart
2	16429085948	1214	1	000002	B2	Standart
3	14945614554	1214	1	000003	B3	Standart
4	11985981251	1214	1	000004	C1	Standart
5	18958959898	1214	1	000005	C2	Standart
6	15620505060	1214	1	000006	C3	Standart
7	15165195818	1214	1	000007	D1	Standart
8	18951951581	1214	1	000008	A1	VIP
9	19848948981	1214	1	000009	A2	VIP
10	11651526516	1214	1	000010	A3	VIP
11	15020955900	1237	2	000011	B1	Standart
12	10020566506	1237	2	000012	B2	Standart
13	10561898914	1237	2	000013	B3	Standart
14	19859819785	1237	2	000014	C1	Standart
15	15814985519	1237	2	000015	C2	Standart

After deletion on Client Table-Cascading effect on Seat Table

	client_id	event_id	venue_id	seat_id	seat_name	seat_type
1	18098680134	1214	1	000001	B1	Standart
2	14945614554	1214	1	000003	B3	Standart
3	11985981251	1214	1	000004	C1	Standart
4	18958959898	1214	1	000005	C2	Standart
5	15620505060	1214	1	000006	C3	Standart
6	15165195818	1214	1	000007	D1	Standart
7	18951951581	1214	1	000008	A1	VIP
8	19848948981	1214	1	000009	A2	VIP
9	11651526516	1214	1	000010	A3	VIP
10	15020955900	1237	2	000011	B1	Standart
11	10020566506	1237	2	000012	B2	Standart
12	10561898914	1237	2	000013	B3	Standart
13	19859819785	1237	2	000014	C1	Standart
14	15814985519	1237	2	000015	C2	Standart
15	15620505060	1237	2	000016	C3	Standart

Update 2: Update existing phone number in Client_PhoneNumber Table. In the event that the customer's recorded phone number is not available (phone malfunction, change of number, etc.), it aims to change the current number with an unavailable phone number. If the customer's more than one phone number is registered, the unavailable number can also be deleted.

update Client_PhoneNumber

set Phone_Number = '+90(5360750505) '

where Phone_Number = '+90(5315159499) ';

Before updating Client_PhoneNumber

	client_id	phone_number
1	18098680134	+90(5315159499)
2	14945614554	+90(5077933353)
3	11985981251	+90(5151596151)
4	18958959898	+90(5186594189)
5	15165195818	+1(5525615618)
6	18951951581	+90(5191591198)
7	19848948981	+90(5436241267)
8	19846516516	+1(5515752757)
9	11651526516	+90(5882743584)
10	15995095950	+90(5589195198)
11	15020955900	+90(5649819519)
12	10020566506	+90(5195151951)
13	10561898914	+90(5551951915)
14	19859819785	+90(5051851918)
15	15814985519	+90(5594148489)

After updating Client_PhoneNumber

	client_id	phone_number
23	14845919514	+90(5324668783)
24	18915615959	+90(5125889485)
25	16518948912	+90(5072519515)
26	16518948912	+90(5324668783)
27	18561591599	+90(5165158851)
28	12059591958	+90(5564565151)
29	12059591958	+90(5148951911)
30	18956159198	+76(5579619515)
31	17988948919	+90(5581561651)
32	10259059005	+99(5165165210)
33	13473735750	+2(5558948515)
34	13068841891	+5(5849851562)
35	16841818154	+15(5165161554)
36	16841818177	+27(5444894515)
37	18098680134	+90(5360750505)

Delete 2: Delete the **phone number** requested by the customer.

delete from Client_PhoneNumber

where Phone_Number = '+90(5345654765)';

Before deletion on Client_PhoneNumber

	client_id	phone_number
24	18915615959	+90(5125889485)
25	16518948912	+90(5072519515)
26	16518948912	+90(5324668783)
27	18561591599	+90(5165158851)
28	11591985191	+90(5345654765)
29	12059591958	+90(5564565151)
30	12059591958	+90(5148951911)
31	18956159198	+76(5579619515)
32	17988948919	+90(5581561651)
33	10259059005	+99(5165165210)
34	13473735750	+2(5558948515)
35	13068841891	+5(5849851562)
36	16841818154	+15(5165161554)
37	16841818177	+27(5444894515)
38	18098680134	+90(5360750505)

After deletion on Client_PhoneNumber

	client_id	phone_number
23	14845919514	+90(5324668783)
24	18915615959	+90(5125889485)
25	16518948912	+90(5072519515)
26	16518948912	+90(5324668783)
27	18561591599	+90(5165158851)
28	12059591958	+90(5564565151)
29	12059591958	+90(5148951911)
30	18956159198	+76(5579619515)
31	17988948919	+90(5581561651)
32	10259059005	+99(5165165210)
33	13473735750	+2(5558948515)
34	13068841891	+5(5849851562)
35	16841818154	+15(5165161554)
36	16841818177	+27(5444894515)
37	18098680134	+90(5360750505)

Delete 2: Delete all tuples in the **Client_PhoneNumber** relation for clients whose phone number's does not start with +90. Due to the high cost of international numbers, it was decided to delete the foreign numbers in order to ensure minimize cost. This customers can be reached via email instead of their phone number.

delete from Client_PhoneNumber

where Phone_Number **not like** '+90%';

Before deletion on Client_PhoneNumber

	client_id	phone_number
1	14945614554	+90(5077933353)
2	11985981251	+90(5151596151)
3	18958959898	+90(5186594189)
4	15165195818	+1(5525615618)
5	18951951581	+90(5191591198)
6	19848948981	+90(5436241267)
7	19846516516	+1(5515752757)
8	11651526516	+90(5882743584)
9	15995095950	+90(5589195198)
10	15020955900	+90(5649819519)
11	10020566506	+90(5195151951)
12	10561898914	+90(5551951915)
13	19859819785	+90(5051851918)
14	15814985519	+90(5594148489)
15	15620505060	+90(5536618519)

After deletion on Client_PhoneNumber

	client_id	phone_number
1	14945614554	+90(5077933353)
2	11985981251	+90(5151596151)
3	18958959898	+90(5186594189)
4	18951951581	+90(5191591198)
5	19848948981	+90(5436241267)
6	11651526516	+90(5882743584)
7	15995095950	+90(5589195198)
8	15020955900	+90(5649819519)
9	10020566506	+90(5195151951)
10	10561898914	+90(5551951915)
11	19859819785	+90(5051851918)
12	15814985519	+90(5594148489)
13	15620505060	+90(5536618519)
14	14065159815	+90(5551541981)
15	17561590505	+90(5568498561)

Now, all client's phone numbers start with +90.

Update 3: Update existing event in Event Table (cascading effects on other tables). When updating Event_ID of Event table, it will automatically update other tables.

update Event

set Event_ID = '9999'

where Event_ID = '1111';

Before updating Event Table

	event_id	name	genre	date_of_start	date_of_end
1	1111	12 Angry Men	Cinema	2020-03-29	2020-03-30
2	1214	Hamlet	Theater	2020-03-23	2020-03-25
3	1224	Revival Tour	Concert	2016-03-23	2020-03-24
4	1981	Europa Galante	Opera	2020-03-23	2020-03-24
5	1676	Harry Potter and the Deathly Hallows – Part 1	Cinema	2020-03-23	2020-03-24
6	1237	Çalikuşu	Theater	2020-03-24	2020-03-25
7	1785	Harry Potter and the Deathly Hallows – Part 2	Cinema	2020-03-25	2020-05-26
8	1573	Tarkan Concert	Concert	2020-03-25	2020-03-27
9	1025	Siyahlı Kadın	Theater	2020-03-26	2020-03-27
10	1783	Otto Barok	Opera	2020-03-27	2020-03-28
11	1023	Star Wars	Cinema	2020-03-25	2020-03-26
12	1092	Pride and Prejudice	Cinema	2020-03-23	2020-03-27
13	1293	Turandot	Opera	2020-03-24	2020-03-27
14	1862	Truva Sonatı	Concert	2020-03-23	2020-03-25
15	1795	Operadaki Hayalet	Opera	2020-03-23	2020-03-25

After updating Event Table

	event_id	name	genre	date_of_start	date_of_end
9	1783	Otto Barok	Opera	2020-03-27	2020-03-28
10	1023	Star Wars	Cinema	2020-03-25	2020-03-26
11	1092	Pride and Prejudice	Cinema	2020-03-23	2020-03-27
12	1293	Turandot	Opera	2020-03-24	2020-03-27
13	1862	Truva Sonatı	Concert	2020-03-23	2020-03-25
14	1795	Operadaki Hayalet	Opera	2020-03-23	2020-03-25
15	9516	Eyvah Nadir	Theater	2020-03-23	2020-03-25
16	9846	Lüküs Hayat	Theater	2020-03-24	2020-03-27
17	1985	The Irishman	Cinema	2020-03-24	2020-03-27
18	7961	Aida	Opera	2020-03-26	2020-03-27
19	5515	Evgeny Grinko	Concert	2020-03-24	2020-03-24
20	9573	Inception	Cinema	2020-03-24	2020-03-26
21	1811	Thelma and Louise	Cinema	2020-03-23	2020-03-25
22	6486	Groundhog Day	Cinema	2020-03-24	2020-03-27
23	9999	12 Angry Men	Cinema	2020-03-29	2020-03-30

Before updating Event Table (Venue Table)

	venueeventid	venue_id	vphone_number	name	section	information
1	1214	1	3122400091	Cüneyt Gökçer Stage	Hall 1	https://tiyatrol.com
2	1237	2	3122400091	Cüneyt Gökçer Stage	Hall 2	https://tiyatrol.com
3	1025	3	3122400091	Cüneyt Gökçer Stage	Hall 3	https://tiyatrol.com
4	9516	4	3124671744	Şinasi Stage	Hall 1	https://tiyatrol.com
5	9846	5	3124671744	Şinasi Stage	Hall 2	https://tiyatrol.com
6	1981	6	2125722070	Leyla Gencer Opera and Art Center	Hall 1	http://www.leyla.com
7	1783	7	2125722070	Leyla Gencer Opera and Art Center	Hall 2	http://www.leyla.com
8	1293	8	3123242214	Ankara General Directorate of Opera and Ba	Hall 1	https://www.oq.com
9	1795	9	3123242214	Ankara General Directorate of Opera and Ba	Hall 2	https://www.oq.com
10	7961	10	3123242214	Ankara General Directorate of Opera and Ba	Hall 3	https://www.oq.com
11	1111	11	3122411241	Arcadium Cinema	Hall 1	http://www.arcadium.com
12	1676	12	3122411241	Arcadium Cinema	Hall 2	http://www.arcadium.com
13	1785	13	3122199393	Prestige Cinema	Hall 1	http://www.prestige.com
14	1023	14	8502200967	Cinemaximum AnkaMall	Hall 1	https://www.cinemaximum.com

After updating Event Table-cascading effect on Venue Table

	venueeventid	venue_id	vphone_number	name	section	information
9	1795	9	3123242214	Ankara General Directorate of Opera and Ba	Hall 2	https://www.oq.com
10	7961	10	3123242214	Ankara General Directorate of Opera and Ba	Hall 3	https://www.oq.com
11	1676	12	3122411241	Arcadium Cinema	Hall 2	http://www.arcadium.com
12	1785	13	3122199393	Prestige Cinema	Hall 1	http://www.prestige.com
13	1023	14	8502200967	Cinemaximum AnkaMall	Hall 1	https://www.cinemaximum.com
14	1092	15	2124813383	Cinemaximum AnkaMall	Hall 2	https://www.cinemaximum.com
15	1985	16	2124813383	Cinemaximum AnkaMall	Hall 3	https://www.cinemaximum.com
16	9573	17	2124813383	Metropol Cinema	Hall 6	https://www.metropol.com
17	1224	18	3122850385	Congresium Kongre ve Sergi Merkezi	Main Section	https://www.congresium.com
18	1573	19	2129123776	Zorlu Center Performing Arts Center	Main Section	https://www.zorlu.com
19	1862	20	8505490126	Jolly Joker	Main Section	https://jollyjoker.com
20	5515	21	2129123776	Babylon	Main Section	https://babylon.com
21	9999	11	3122411241	Arcadium Cinema	Hall 1	http://www.arcadium.com

Delete 3: When an event is deleted, referenced rows of other tables are also deleted automatically (cascading effects on other tables). In case the event is canceled due to some problems, that event and everything referencing that event should be deleted.

delete from Event

where Event_ID = '9999';

Before deletion on Event Table

	event_id	name	genre	date_of_start	date_of_end
9	1783	Otto Barok	Opera	2020-03-27	2020-03-28
10	1023	Star Wars	Cinema	2020-03-25	2020-03-26
11	1092	Pride and Prejudice	Cinema	2020-03-23	2020-03-27
12	1293	Turandot	Opera	2020-03-24	2020-03-27
13	1862	Truva Sonatı	Concert	2020-03-23	2020-03-25
14	1795	Operadaki Hayalet	Opera	2020-03-23	2020-03-25
15	9516	Eyvah Nadir	Theater	2020-03-23	2020-03-25
16	9846	Lüküs Hayat	Theater	2020-03-24	2020-03-27
17	1985	The Irishman	Cinema	2020-03-24	2020-03-27
18	7961	Aida	Opera	2020-03-26	2020-03-27
19	5515	Evgeny Grinko	Concert	2020-03-24	2020-03-24
20	9573	Inception	Cinema	2020-03-24	2020-03-26
21	1811	Thelma and Louise	Cinema	2020-03-23	2020-03-25
22	6486	Groundhog Day	Cinema	2020-03-24	2020-03-27
23	9999	12 Angry Men	Cinema	2020-03-29	2020-03-30

After deletion on Event Table

	event_id	name	genre	date_of_start	date_of_end
8	1025	Siyahlı Kadın	Theater	2020-03-26	2020-03-27
9	1783	Otto Barok	Opera	2020-03-27	2020-03-28
10	1023	Star Wars	Cinema	2020-03-25	2020-03-26
11	1092	Pride and Prejudice	Cinema	2020-03-23	2020-03-27
12	1293	Turandot	Opera	2020-03-24	2020-03-27
13	1862	Truva Sonatı	Concert	2020-03-23	2020-03-25
14	1795	Operadaki Hayalet	Opera	2020-03-23	2020-03-25
15	9516	Eyvah Nadir	Theater	2020-03-23	2020-03-25
16	9846	Lüküs Hayat	Theater	2020-03-24	2020-03-27
17	1985	The Irishman	Cinema	2020-03-24	2020-03-27
18	7961	Aida	Opera	2020-03-26	2020-03-27
19	5515	Evgeny Grinko	Concert	2020-03-24	2020-03-24
20	9573	Inception	Cinema	2020-03-24	2020-03-26
21	1811	Thelma and Louise	Cinema	2020-03-23	2020-03-25
22	6486	Groundhog Day	Cinema	2020-03-24	2020-03-27

Before deletion on Event Table (Venue Table)

	venueeventid	venue_id	vphone_number	abc name	abc section	abc information
9	1795	9	3123242214	Ankara General Directorate of Opera and Ba	Hall 2	https://www.
10	7961	10	3123242214	Ankara General Directorate of Opera and Ba	Hall 3	https://www.
11	1676	12	3122411241	Arcadium Cinema	Hall 2	http://www.a
12	1785	13	3122199393	Prestige Cinema	Hall 1	http://www.ç
13	1023	14	8502200967	Cinemaximum AnkaMall	Hall 1	https://www.
14	1092	15	2124813383	Cinemaximum AnkaMall	Hall 2	https://www.
15	1985	16	2124813383	Cinemaximum AnkaMall	Hall 3	https://www.
16	9573	17	2124813383	Metropol Cinema	Hall 6	https://www.
17	1224	18	3122850385	Congresium Kongre ve Sergi Merkezi	Main Section	https://www.
18	1573	19	2129123776	Zorlu Center Performing Arts Center	Main Section	https://www.
19	1862	20	8505490126	Jolly Joker	Main Section	https://jollyj
20	5515	21	2129123776	Babylon	Main Section	https://babyl
21	9999	11	3122411241	Arcadium Cinema	Hall 1	http://www.a

After deletion on Event Table-cascading effect on Venue Table

	venueeventid	venue_id	vphone_number	abc name	abc section	abc information
8	1293	8	3123242214	Ankara General Directorate of Opera and Ba	Hall 1	https://www.
9	1795	9	3123242214	Ankara General Directorate of Opera and Ba	Hall 2	https://www.
10	7961	10	3123242214	Ankara General Directorate of Opera and Ba	Hall 3	https://www.
11	1676	12	3122411241	Arcadium Cinema	Hall 2	http://www.a
12	1785	13	3122199393	Prestige Cinema	Hall 1	http://www.ç
13	1023	14	8502200967	Cinemaximum AnkaMall	Hall 1	https://www.
14	1092	15	2124813383	Cinemaximum AnkaMall	Hall 2	https://www.
15	1985	16	2124813383	Cinemaximum AnkaMall	Hall 3	https://www.
16	9573	17	2124813383	Metropol Cinema	Hall 6	https://www.
17	1224	18	3122850385	Congresium Kongre ve Sergi Merkezi	Main Section	https://www.
18	1573	19	2129123776	Zorlu Center Performing Arts Center	Main Section	https://www.
19	1862	20	8505490126	Jolly Joker	Main Section	https://jollyj
20	5515	21	2129123776	Babylon	Main Section	https://babyl

Update 4: Update existing email in Cast Table. In order for the people on cast to be accessible, if the data entered for the access changes, the old data entered must be updated.

update Castt

set Email = 'Pianist.Fazil.Say@gmail.com'

where Email = 'fazil.say@gmail.com';

Before updating Cast Table

	casteventid	abc name	abc surname	abc profession	abc email	123 salary
5	1676	Emma	Watson	Actor	emma.watson@gmail.com	12,000
6	1676	Rupertz	Grint	Actor	rupertz.grint@gmail.com	12,000
7	1237	Nur	Yazar	Actor	nur.yazar@gmail.com	7,000
8	1237	Sinem	Şahin	Actor	sinem.sahin@gmail.com	7,000
9	1237	E. Çağrı	Turan	Actor	cagri.turan@gmail.com	7,000
10	1785	Daniel	Ratcliffe	Actor	daniel.ratcliffe@gmail.com	12,000
11	1785	Emma	Watson	Actor	emma.watson@gmail.com	12,000
12	1785	Rupertz	Grint	Actor	rupertz.grint@gmail.com	12,000
13	1573	Tarkan	Tevetoğlu	Singer	tarkan@gmail.com	950,000
14	1025	Mazda	Özdemir	Actor	m.ozdemir@gmail.com	8,700
15	1783	Orhan	Avcı	Chef and Violinist	orhan.avci@gmail.com	7,800
16	1023	Carrie	Fisher	Actor	carrie.fisher@gmail.com	86,000
17	1092	Keira	Knightley	Actor	keira.knightley@gmail.com	234,000
18	1293	Franco	Alfano	Composer	falfano@gmail.com	39,000
19	1862	Fazil	Say	Pianist	fazil.say@gmail.com	340,000

After updating Cast Table

	casteventid	abc name	abc surname	abc profession	abc email	123 salary
18	1293	Franco	Alfano	Composer	falfano@gmail.com	39,000
19	1795	Caston	Leroux	Singer	casleroux@gmail.com	283,000
20	9516	Koray	Karaca	Actor	kkaraca@gmail.com	495,000
21	9846	Mehmet Ali	Toklu	Actor	matoklu@gmail.com	49,000
22	1985	Al	Pacino	Actor	alpacino@gmail.com	584,000
23	7961	Antonio	Ghislanzoni	Composer	aghislanzoni@gmail.com	599,000
24	5515	Yevgeny	Grinko	Pianist	yevgrinko@gmail.com	602,000
25	9573	Leonardo	Dicaprio	Actor	leo.dicaprio@gmail.com	959,000
26	1811	Susan	Sarandon	Actor	ssarandon@gmail.com	[NULL]
27	1811	Geena	Davis	Actor	geena.davis@gmail.com	[NULL]
28	1811	Brad	Pitt	Actor	brad.pitt@gmail.com	[NULL]
29	6486	Bill	Murray	Actor	bill.murray@gmail.com	[NULL]
30	6486	Andie	MacDowell	Actor	andie.macdowell@gmail.co	[NULL]
31	6486	Chris	Elliott	Actor	chris.elliott@gmail.com	[NULL]
32	1862	Fazil	Say	Pianist	Pianist.Fazil.Say@gmail.com	340,000

Delete 4: Delete tuple in **Cast Table**. If the person in the cast leaves the event, the actor, composer, singer etc. must be deleted in that event.

delete from Castt

where email = 'falfano@gmail.com';

Before deletion on Cast Table

	casteventid	abc name	abc surname	abc profession	email	salary
12	1785	Rupertz	Grint	Actor	rupertz.grint@gmail.com	12,000
13	1573	Tarkan	Tevetoğlu	Singer	tarkan@gmail.com	950,000
14	1025	Mazda	Özdemir	Actor	m.ozdemir@gmail.com	8,700
15	1783	Orhan	Avci	Chef and Violinist	orhan.avci@gmail.com	7,800
16	1023	Carrie	Fisher	Actor	carrie.fisher@gmail.com	86,000
17	1092	Keira	Knightley	Actor	keira.knightley@gmail.com	234,000
18	1293	Franco	Alfano	Composer	falfano@gmail.com	39,000
19	1795	Caston	Leroux	Singer	casleroux@gmail.com	283,000
20	9516	Koray	Karaca	Actor	kkaraca@gmail.com	495,000
21	9846	Mehmet Ali	Toklu	Actor	matoklu@gmail.com	49,000
22	1985	Al	Pacino	Actor	alpacino@gmail.com	584,000
23	7961	Antonio	Ghislanzoni	Composer	aghislanzoni@gmail.com	599,000
24	5515	Yevgeny	Grinko	Pianist	yevgrinko@gmail.com	602,000
25	9573	Leonardo	Dicaprio	Actor	leo.dicaprio@gmail.com	959,000
26	1811	Susan	Sarandon	Actor	ssarandon@gmail.com	[NULL]

After deletion on Cast Table

	casteventid	abc name	abc surname	abc profession	email	salary
11	1785	Emma	Watson	Actor	emma.watson@gmail.com	12,000
12	1785	Rupertz	Grint	Actor	rupertz.grint@gmail.com	12,000
13	1573	Tarkan	Tevetoğlu	Singer	tarkan@gmail.com	950,000
14	1025	Mazda	Özdemir	Actor	m.ozdemir@gmail.com	8,700
15	1783	Orhan	Avci	Chef and Violinist	orhan.avci@gmail.com	7,800
16	1023	Carrie	Fisher	Actor	carrie.fisher@gmail.com	86,000
17	1092	Keira	Knightley	Actor	keira.knightley@gmail.com	234,000
18	1795	Caston	Leroux	Singer	casleroux@gmail.com	283,000
19	9516	Koray	Karaca	Actor	kkaraca@gmail.com	495,000
20	9846	Mehmet Ali	Toklu	Actor	matoklu@gmail.com	49,000
21	1985	Al	Pacino	Actor	alpacino@gmail.com	584,000
22	7961	Antonio	Ghislanzoni	Composer	aghislanzoni@gmail.com	599,000
23	5515	Yevgeny	Grinko	Pianist	yevgrinko@gmail.com	602,000
24	9573	Leonardo	Dicaprio	Actor	leo.dicaprio@gmail.com	959,000
25	1811	Susan	Sarandon	Actor	ssarandon@gmail.com	[NULL]

Update 5: Update existing given score of event in **Event Review Table**. The maximum score is determined out of 10. For this reason, the score of customers with a score greater than 10 is updated as 10.

update Event_Review

set Given_Score = '10'

where Given_Score > '10');

Before updating Event Review Table

	revieweventid	evaluating_client_id	evaluating_date	given_score
32	1023	17818495050	2020-05-17	7
33	1224	17818495050	2020-05-18	90
34	1785	17818495050	2020-06-20	9
35	1676	17115961595	2020-06-29	9
36	1783	17115961595	2020-06-25	6
37	1785	17115961595	2020-07-13	10
38	1224	17115961595	2020-08-30	9.5
39	1023	17115961595	2020-09-30	7
40	1237	15165195818	2020-11-17	7
41	9573	15165195818	2020-12-17	9.5
42	1023	15165195818	2020-07-18	6.9
43	1862	10259059005	2020-09-02	2
44	1795	18561591599	2020-07-15	44.5
45	1785	18561591599	2020-07-16	5
46	1795	16518948912	2020-10-10	5.5

More than 10 points entered for Revival Tour (1224) and Operadaki Hayalet (1795).

After updating Event Review Table

	revieweventid	evaluating_client_id	evaluating_date	given_score
36	1224	17115961595	2020-08-30	9.5
37	1023	17115961595	2020-09-30	7
38	1237	15165195818	2020-11-17	7
39	9573	15165195818	2020-12-17	9.5
40	1023	15165195818	2020-07-18	6.9
41	1862	10259059005	2020-09-02	2
42	1785	18561591599	2020-07-16	5
43	1795	16518948912	2020-10-10	5.5
44	1785	16518948912	2020-11-16	7.5
45	1676	18789195516	2020-02-07	8
46	1237	18789195516	2020-02-25	7
47	1224	16841818154	2020-12-11	10
48	1224	17818495050	2020-05-18	10
49	1795	18561591599	2020-07-15	10
50	1785	18789195516	2020-05-17	10

The edited version of scores with more than 10 points.

Delete 5: Delete tuple in Event Review table. If the client wants to delete his/her review, Evaluating_Clint_ID should be deleted in Event Review Table.

delete from Event_Review

where Evaluating_Client_ID= '14945614554';

Before deletion on Event Review Table

	revieweventid	evaluating_client_id	evaluating_date	given_score
1	1676	18098680134	2020-06-17	9
2	1785	18098680134	2020-08-10	8
3	1224	18098680134	2020-05-17	10
4	1224	16429085948	2020-05-16	10
5	1676	16429085948	2020-11-19	9.5
6	1214	14945614554	2020-08-15	5
7	1981	14845919514	2020-12-30	2
8	1981	17115961595	2020-03-11	5
9	1785	12059591958	2020-04-03	8
10	1676	15620505060	2020-09-17	8
11	1224	11651526516	2020-05-25	9.9
12	1785	10020566506	2020-05-26	7.9
13	1676	10020566506	2020-11-01	8.5
14	1573	11985981251	2020-08-08	4
15	1573	12059591958	2020-08-08	5.5

After deletion on Event Review Table

	revieweventid	evaluating_client_id	evaluating_date	given_score
1	1676	18098680134	2020-06-17	9
2	1785	18098680134	2020-08-10	8
3	1224	18098680134	2020-05-17	10
4	1224	16429085948	2020-05-16	10
5	1676	16429085948	2020-11-19	9.5
6	1981	14845919514	2020-12-30	2
7	1981	17115961595	2020-03-11	5
8	1785	12059591958	2020-04-03	8
9	1676	15620505060	2020-09-17	8
10	1224	11651526516	2020-05-25	9.9
11	1785	10020566506	2020-05-26	7.9
12	1676	10020566506	2020-11-01	8.5
13	1573	11985981251	2020-08-08	4
14	1573	12059591958	2020-08-08	5.5
15	1023	19859819785	2020-07-17	8

Update 6: Update existing venue in **Venue Table** (cascading effects on other tables). When updating Venue_ID of Venue table, it will automatically update other tables. If the place where the event will take place changes, the new venue needs to be updated.

update Venue

set Venue_ID = '888'

where Venue_ID = '8';

Before updating Venue Table

	venueeventid	venue_id	vphone_number	name	section	information
1	1214	1	3122400091	Cüneyt Gökçer Stage	Hall 1	https://tiyatrolar.com
2	1237	2	3122400091	Cüneyt Gökçer Stage	Hall 2	https://tiyatrolar.com
3	1025	3	3122400091	Cüneyt Gökçer Stage	Hall 3	https://tiyatrolar.com
4	9516	4	3124671744	Şinasi Stage	Hall 1	https://tiyatrolar.com
5	9846	5	3124671744	Şinasi Stage	Hall 2	https://tiyatrolar.com
6	1981	6	2125722070	Leyla Gencer Opera and Art Center	Hall 1	http://www.leylagenceri.com
7	1783	7	2125722070	Leyla Gencer Opera and Art Center	Hall 2	http://www.leylagenceri.com
8	1293	8	3123242214	Ankara General Directorate of Opera and Ba	Hall 1	https://www.operabiz.com
9	1795	9	3123242214	Ankara General Directorate of Opera and Ba	Hall 2	https://www.operabiz.com
10	7961	10	3123242214	Ankara General Directorate of Opera and Ba	Hall 3	https://www.operabiz.com
11	1676	12	3122411241	Arcadium Cinema	Hall 2	http://www.arcadium.com
12	1785	13	3122199393	Prestige Cinema	Hall 1	http://www.prestige.com
13	1023	14	8502200967	Cinemaximum AnkaMall	Hall 1	https://www.cinemaximum.com
14	1092	15	2124813383	Cinemaximum AnkaMall	Hall 2	https://www.cinemaximum.com
15	1985	16	2124813383	Cinemaximum AnkaMall	Hall 3	https://www.cinemaximum.com

After updating Venue Table

	venueeventid	venue_id	vphone_number	name	section	information
7	1783	7	2125722070	Leyla Gencer Opera and Art Center	Hall 2	http://www.leylagenceri.com
8	1795	9	3123242214	Ankara General Directorate of Opera and Ba	Hall 2	https://www.operabiz.com
9	7961	10	3123242214	Ankara General Directorate of Opera and Ba	Hall 3	https://www.operabiz.com
10	1676	12	3122411241	Arcadium Cinema	Hall 2	http://www.arcadium.com
11	1785	13	3122199393	Prestige Cinema	Hall 1	http://www.prestige.com
12	1023	14	8502200967	Cinemaximum AnkaMall	Hall 1	https://www.cinemaximum.com
13	1092	15	2124813383	Cinemaximum AnkaMall	Hall 2	https://www.cinemaximum.com
14	1985	16	2124813383	Cinemaximum AnkaMall	Hall 3	https://www.cinemaximum.com
15	9573	17	2124813383	Metropol Cinema	Hall 6	https://www.avarsin.com
16	1224	18	3122850385	Congresium Kongre ve Sergi Merkezi	Main Section	https://www.cinemaximum.com
17	1573	19	2129123776	Zorlu Center Performing Arts Center	Main Section	https://www.zorlucenter.com
18	1862	20	8505490126	Jolly Joker	Main Section	https://www.jollyjoker.com
19	5515	21	2129123776	Babylon	Main Section	https://www.babylon.com
20	1293	888	3123242214	Ankara General Directorate of Opera and Ba	Hall 1	https://www.operabiz.com

Before updating Venue Table (Seat Table)

	client_id	event_id	venue_id	seat_id	seat_name	seat_type
64	12059591958	1783	7	000067	D1	Standart
65	10020566506	1783	7	000068	A1	VIP
66	10561898914	1783	7	000069	A2	VIP
67	19859819785	1783	7	000070	A3	VIP
68	15814985519	1293	8	000071	B1	Standart
69	18958959898	1293	8	000072	B2	Standart
70	15165195818	1293	8	000073	B3	Standart
71	18951951581	1293	8	000074	C1	Standart
72	19848948981	1293	8	000075	C2	Standart
73	17818495050	1293	8	000076	C3	Standart
74	14845919514	1293	8	000077	D1	Standart
75	18915615959	1293	8	000078	A1	VIP
76	11591985191	1293	8	000079	A2	VIP
77	12059591958	1293	8	000080	A3	VIP
78	14945614554	1795	9	000081	B1	Standart
79	11985981251	1795	9	000082	B2	Standart

After updating Venue Table-cascading effect on Seat Table

	client_id	event_id	venue_id	seat_id	seat_name	seat_type
95	18789195516	1676	12	000118	A1	Standart
96	11651526516	1676	12	000119	A2	Standart
97	14945614554	1676	12	000120	A3	Standart
98	15814985519	1293	888	000071	B1	Standart
99	18958959898	1293	888	000072	B2	Standart
100	15165195818	1293	888	000073	B3	Standart
101	18951951581	1293	888	000074	C1	Standart
102	19848948981	1293	888	000075	C2	Standart
103	17818495050	1293	888	000076	C3	Standart
104	14845919514	1293	888	000077	D1	Standart
105	18915615959	1293	888	000078	A1	VIP
106	11591985191	1293	888	000079	A2	VIP
107	12059591958	1293	888	000080	A3	VIP
108	18098680134	1785	13	000122	B2	Standart
109	18958959898	1785	13	000123	B3	Standart
110	17561590505	1785	13	000124	C1	Standart

Delete 6: When an **venue** is deleted, referenced rows of other tables are also deleted automatically (cascading effects on other tables). If the venue is closed, changed hands or restored, that venue should be deleted. Also, if there is an event registered in that venue, that event's venues must be updated.

delete from Venue

where Venue_ID = '7';

Before deletion on Venue Table

	venueeventid	venue_id	vphone_number	abc name	abc section	abc information
1	1214	1	3122400091	Cüneyt Gökçer Stage	Hall 1	https://tiyatrolar.com
2	1237	2	3122400091	Cüneyt Gökçer Stage	Hall 2	https://tiyatrolar.com
3	1025	3	3122400091	Cüneyt Gökçer Stage	Hall 3	https://tiyatrolar.com
4	9516	4	3124671744	Şinasi Stage	Hall 1	https://tiyatrolar.com
5	9846	5	3124671744	Şinasi Stage	Hall 2	https://tiyatrolar.com
6	1981	6	2125722070	Leyla Gencer Opera and Art Center	Hall 1	http://www.leylageni.com
7	1783	7	2125722070	Leyla Gencer Opera and Art Center	Hall 2	http://www.leylageni.com
8	1795	9	3123242214	Ankara General Directorate of Opera and Ba	Hall 2	https://www.operabi.com
9	7961	10	3123242214	Ankara General Directorate of Opera and Ba	Hall 3	https://www.operabi.com
10	1676	12	3122411241	Arcadium Cinema	Hall 2	http://www.arcadium.com
11	1785	13	3122199393	Prestige Cinema	Hall 1	http://www.prestige.com
12	1023	14	8502200967	Cinemaximum AnkaMall	Hall 1	https://www.cinema.com
13	1092	15	2124813383	Cinemaximum AnkaMall	Hall 2	https://www.cinema.com
14	1985	16	2124813383	Cinemaximum AnkaMall	Hall 3	https://www.cinema.com
15	9573	17	2124813383	Metropol Cinema	Hall 6	https://www.avarsin.com

After deletion on Venue Table

	venueeventid	venue_id	vphone_number	abc name	abc section	abc information
1	1214	1	3122400091	Cüneyt Gökçer Stage	Hall 1	https://tiyatrolar.com
2	1237	2	3122400091	Cüneyt Gökçer Stage	Hall 2	https://tiyatrolar.com
3	1025	3	3122400091	Cüneyt Gökçer Stage	Hall 3	https://tiyatrolar.com
4	9516	4	3124671744	Şinasi Stage	Hall 1	https://tiyatrolar.com
5	9846	5	3124671744	Şinasi Stage	Hall 2	https://tiyatrolar.com
6	1981	6	2125722070	Leyla Gencer Opera and Art Center	Hall 1	http://www.leylageni.com
7	1795	9	3123242214	Ankara General Directorate of Opera and Ba	Hall 2	https://www.operabi.com
8	7961	10	3123242214	Ankara General Directorate of Opera and Ba	Hall 3	https://www.operabi.com
9	1676	12	3122411241	Arcadium Cinema	Hall 2	http://www.arcadium.com
10	1785	13	3122199393	Prestige Cinema	Hall 1	http://www.prestige.com
11	1023	14	8502200967	Cinemaximum AnkaMall	Hall 1	https://www.cinema.com
12	1092	15	2124813383	Cinemaximum AnkaMall	Hall 2	https://www.cinema.com
13	1985	16	2124813383	Cinemaximum AnkaMall	Hall 3	https://www.cinema.com
14	9573	17	2124813383	Metropol Cinema	Hall 6	https://www.avarsin.com
15	1224	18	3122850385	Congresium Kongre ve Sergi Merkezi	Main Section	https://www.cinema.com

Before deletion on Venue Table (Seat Table)

	client_id	event_id	venue_id	seat_id	seat_name	seat_type
56	17115961595	1981	6	000058	A1	VIP
57	17818495050	1981	6	000059	A2	VIP
58	14845919514	1981	6	000060	A3	VIP
59	18098680134	1783	7	000061	B1	Standart
60	11985981251	1783	7	000063	B3	Standart
61	18951951581	1783	7	000064	C1	Standart
62	17988948919	1783	7	000065	C2	Standart
63	18956159198	1783	7	000066	C3	Standart
64	12059591958	1783	7	000067	D1	Standart
65	10020566506	1783	7	000068	A1	VIP
66	10561898914	1783	7	000069	A2	VIP
67	19859819785	1783	7	000070	A3	VIP
68	14945614554	1795	9	000081	B1	Standart
69	11985981251	1795	9	000082	B2	Standart
70	18951951581	1795	9	000083	B3	Standart
71	10561898914	1795	9	000084	C1	Standart

After deletion on Venue Table-cascading effect on Seat Table

	client_id	event_id	venue_id	seat_id	seat_name	seat_type
52	14065159815	1981	6	000054	C1	Standart
53	17561590505	1981	6	000055	C2	Standart
54	18789195516	1981	6	000056	C3	Standart
55	16059159050	1981	6	000057	D1	Standart
56	17115961595	1981	6	000058	A1	VIP
57	17818495050	1981	6	000059	A2	VIP
58	14845919514	1981	6	000060	A3	VIP
59	14945614554	1795	9	000081	B1	Standart
60	11985981251	1795	9	000082	B2	Standart
61	18951951581	1795	9	000083	B3	Standart
62	10561898914	1795	9	000084	C1	Standart
63	19859819785	1795	9	000085	C2	Standart
64	18098680134	1795	9	000086	C3	Standart
65	12059591958	1795	9	000087	D1	Standart
66	11591985191	1795	9	000088	A1	VIP
67	16518948912	1795	9	000089	A2	VIP

Update 7: Update existing seat in **Seat Table**. In case seats are added to the lounge or the seat order changes, the seat numbers must be updated. When updating Seat_ID of Seat table, it will automatically update other tables (cascading effects on other tables).

update Seat

set Seat_ID = '111111'

where Seat_ID = '000001';

Before updating Seat Table

	client_id	event_id	venue_id	seat_id	seat_name	seat_type
1	18098680134	1214	1	000001	B1	Standart
2	14945614554	1214	1	000003	B3	Standart
3	11985981251	1214	1	000004	C1	Standart
4	18958959898	1214	1	000005	C2	Standart
5	15620505060	1214	1	000006	C3	Standart
6	15165195818	1214	1	000007	D1	Standart
7	18951951581	1214	1	000008	A1	VIP
8	19848948981	1214	1	000009	A2	VIP
9	11651526516	1214	1	000010	A3	VIP
10	15020955900	1237	2	000011	B1	Standart
11	10020566506	1237	2	000012	B2	Standart
12	10561898914	1237	2	000013	B3	Standart
13	19859819785	1237	2	000014	C1	Standart
14	15814985519	1237	2	000015	C2	Standart
15	15620505060	1237	2	000016	C3	Standart
16	14065159815	1237	2	000017	D1	Standart

After updating Seat Table

	client_id	event_id	venue_id	seat_id	seat_name	seat_type
94	14845919514	1293	888	000077	D1	Standart
95	18915615959	1293	888	000078	A1	VIP
96	11591985191	1293	888	000079	A2	VIP
97	12059591958	1293	888	000080	A3	VIP
98	18098680134	1214	1	111111	B1	Standart
99	18098680134	1785	13	000122	B2	Standart
100	18958959898	1785	13	000123	B3	Standart
101	17561590505	1785	13	000124	C1	Standart
102	12059591958	1785	13	000125	C2	Standart
103	14845919514	1785	13	000126	C3	Standart
104	15620505060	1785	13	000127	D1	Standart
105	15995095950	1785	13	000128	A1	Standart
106	17988948919	1785	13	000129	A2	Standart
107	15020955900	1785	13	000130	A3	Standart
108	14845919514	1023	14	000131	B1	Standart
109	17561590505	1023	14	000132	B2	Standart

Before updating Seat Table (Seat Review Table)

	client_id	event_id	venue_id	seat_id	review_id	seat_angle	seat_image	comment_date
1	18098680134	1214	1	000001	000001	5		1995-04-07
2	14945614554	1214	1	000003	000003	5		1995-04-07
3	11985981251	1214	1	000004	000004	10		1995-04-07
4	18958959898	1214	1	000005	000005	0		1995-04-07
5	15620505060	1214	1	000006	000006	10		1995-04-07
6	15165195818	1214	1	000007	000007	0		1995-04-07
7	18951951581	1214	1	000008	000008	0		1995-04-07
8	19848948981	1214	1	000009	000009	0		1995-04-07
9	11651526516	1214	1	000010	000010	0		1995-04-07
10	15020955900	1237	2	000011	000011	5		1995-04-07
11	10020566506	1237	2	000012	000012	0		1995-04-07
12	10561898914	1237	2	000013	000013	5		1995-04-07
13	19859819785	1237	2	000014	000014	10		1995-04-07
14	15814985519	1237	2	000015	000015	0		1995-04-07
15	15620505060	1237	2	000016	000016	10		1995-04-07
16	14065159815	1237	2	000017	000017	0		1995-04-07

After updating Seat Table-cascading effect on Seat Review Table

	client_id	event_id	venue_id	seat_id	review_id	seat_angle	seat_image	comment_date
78	15814985519	1293	888	000071	000071	5		1948-01-01
79	18958959898	1293	888	000072	000072	0		1948-01-01
80	15165195818	1293	888	000073	000073	5		1948-01-01
81	18951951581	1293	888	000074	000074	10		1948-01-01
82	19848948981	1293	888	000075	000075	0		1948-01-01
83	17818495050	1293	888	000076	000076	10		1948-01-01
84	14845919514	1293	888	000077	000077	0		1948-01-01
85	18915615959	1293	888	000078	000078	0		1948-01-01
86	18098680134	1214	1	111111	000001	5		1995-04-07
87	10561898914	1676	12	000111	000111	5		2003-01-01
88	15814985519	1676	12	000112	000112	0		2003-01-01
89	16059159050	1676	12	000113	000113	5		2003-01-01
90	17818495050	1676	12	000114	000114	10		2003-01-01
91	14845919514	1676	12	000115	000115	0		2003-01-01
92	11591985191	1676	12	000116	000116	10		2003-01-01
93	12059591958	1676	12	000117	000117	0		2003-01-01

Delete 7: When an **seat** is deleted, referenced rows of other tables are also deleted automatically (cascading effects on other tables). If the area is reduced and the seat is removed, the removed seat must be deleted.

delete from Seat

where Seat_ID = '000010';

Before deletion on Seat Table

	client_id	event_id	venue_id	seat_id	seat_name	seat_type
1	14945614554	1214	1	000003	B3	Standart
2	11985981251	1214	1	000004	C1	Standart
3	18958959898	1214	1	000005	C2	Standart
4	15620505060	1214	1	000006	C3	Standart
5	15165195818	1214	1	000007	D1	Standart
6	18951951581	1214	1	000008	A1	VIP
7	19848948981	1214	1	000009	A2	VIP
8	11651526516	1214	1	000010	A3	VIP
9	15020955900	1237	2	000011	B1	Standart
10	10020566506	1237	2	000012	B2	Standart
11	10561898914	1237	2	000013	B3	Standart
12	19859819785	1237	2	000014	C1	Standart
13	15814985519	1237	2	000015	C2	Standart
14	15620505060	1237	2	000016	C3	Standart
15	14065159815	1237	2	000017	D1	Standart

After deletion on Seat Table

	client_id	event_id	venue_id	seat_id	seat_name	seat_type
1	14945614554	1214	1	000003	B3	Standart
2	11985981251	1214	1	000004	C1	Standart
3	18958959898	1214	1	000005	C2	Standart
4	15620505060	1214	1	000006	C3	Standart
5	15165195818	1214	1	000007	D1	Standart
6	18951951581	1214	1	000008	A1	VIP
7	19848948981	1214	1	000009	A2	VIP
8	15020955900	1237	2	000011	B1	Standart
9	10020566506	1237	2	000012	B2	Standart
10	10561898914	1237	2	000013	B3	Standart
11	19859819785	1237	2	000014	C1	Standart
12	15814985519	1237	2	000015	C2	Standart
13	15620505060	1237	2	000016	C3	Standart
14	14065159815	1237	2	000017	D1	Standart
15	17561590505	1237	2	000018	A1	VIP

Before deletion on Seat Table (Seat Review Table)

	client_id	event_id	venue_id	seat_id	review_id	seat_angle	seat_image	comment_date
1	14945614554	1214	1	000003	000003	5		1995-04-07
2	11985981251	1214	1	000004	000004	10		1995-04-07
3	18958959898	1214	1	000005	000005	0		1995-04-07
4	15620505060	1214	1	000006	000006	10		1995-04-07
5	15165195818	1214	1	000007	000007	0		1995-04-07
6	18951951581	1214	1	000008	000008	0		1995-04-07
7	19848948981	1214	1	000009	000009	0		1995-04-07
8	11651526516	1214	1	000010	000010	0		1995-04-07
9	15020955900	1237	2	000011	000011	5		1995-04-07
10	10020566506	1237	2	000012	000012	0		1995-04-07
11	10561898914	1237	2	000013	000013	5		1995-04-07
12	19859819785	1237	2	000014	000014	10		1995-04-07
13	15814985519	1237	2	000015	000015	0		1995-04-07
14	15620505060	1237	2	000016	000016	10		1995-04-07
15	14065159815	1237	2	000017	000017	0		1995-04-07

After deletion on Seat Table-cascading effect on Seat Review Table

	client_id	event_id	venue_id	seat_id	review_id	seat_angle	seat_image	comment_date
1	14945614554	1214	1	000003	000003	5		1995-04-07
2	11985981251	1214	1	000004	000004	10		1995-04-07
3	18958959898	1214	1	000005	000005	0		1995-04-07
4	15620505060	1214	1	000006	000006	10		1995-04-07
5	15165195818	1214	1	000007	000007	0		1995-04-07
6	18951951581	1214	1	000008	000008	0		1995-04-07
7	19848948981	1214	1	000009	000009	0		1995-04-07
8	15020955900	1237	2	000011	000011	5		1995-04-07
9	10020566506	1237	2	000012	000012	0		1995-04-07
10	10561898914	1237	2	000013	000013	5		1995-04-07
11	19859819785	1237	2	000014	000014	10		1995-04-07
12	15814985519	1237	2	000015	000015	0		1995-04-07
13	15620505060	1237	2	000016	000016	10		1995-04-07
14	14065159815	1237	2	000017	000017	0		1995-04-07
15	17561590505	1237	2	000018	000018	0		1995-04-07

Update 8: Update existing review in **Seat Review Table**. Nowadays, the customer has the opportunity to examine the picture of the seat before buying a ticket for the seat. Remaining the old picture of the seat whose place or feature has changed causes insecurity for the customer. Hence, it should be updated. **Comment date** also should be updated when new seat image is uploaded to inform the clients about the date of images.

update Seat_Review

set Comment_Date = '2020-01-01'

where Venue_ID = '1';

Before updating Seat Review Table

	client_id	event_id	venue_id	seat_id	review_id	seat_angle	seat_image	comment_date
1	14945614554	1214	1	000003	000003	5		1995-04-07
2	11985981251	1214	1	000004	000004	10		1995-04-07
3	18958959898	1214	1	000005	000005	0		1995-04-07
4	15620505060	1214	1	000006	000006	10		1995-04-07
5	15165195818	1214	1	000007	000007	0		1995-04-07
6	18951951581	1214	1	000008	000008	0		1995-04-07
7	19848948981	1214	1	000009	000009	0		1995-04-07
8	15020055900	1237	2	000011	000011	5		1995-04-07
9	10020566506	1237	2	000012	000012	0		1995-04-07
10	10561898914	1237	2	000013	000013	5		1995-04-07
11	19859819785	1237	2	000014	000014	10		1995-04-07
12	15814985519	1237	2	000015	000015	0		1995-04-07
13	15620505060	1237	2	000016	000016	10		1995-04-07
14	14065159815	1237	2	000017	000017	0		1995-04-07
15	17561590505	1237	2	000018	000018	0		1995-04-07

After updating Seat Review Table

	client_id	event_id	venue_id	seat_id	review_id	seat_angle	seat_image	comment_date
76	14845919514	1293	888	000077	000077	0		1948-01-01
77	10015615950	1293	888	000078	000078	0		1948-01-01
78	14945614554	1214	1	000003	000003	5		2020-01-01
79	11985981251	1214	1	000004	000004	10		2020-01-01
80	18958959898	1214	1	000005	000005	0		2020-01-01
81	15620505060	1214	1	000006	000006	10		2020-01-01
82	15165195818	1214	1	000007	000007	0		2020-01-01
83	18951951581	1214	1	000008	000008	0		2020-01-01
84	19848948981	1214	1	000009	000009	0		2020-01-01
85	18098680134	1214	1	111111	000001	5		2020-01-01
86	10561898914	1676	12	000111	000111	5		2003-01-01
87	15814985519	1676	12	000112	000112	0		2003-01-01
88	16059159050	1676	12	000113	000113	5		2003-01-01
89	17818495050	1676	12	000114	000114	10		2003-01-01
90	14845919514	1676	12	000115	000115	0		2003-01-01

Delete 8: Delete tuple in **Seat Review Table**. If the business owner requests a copyright in the seat review due to the strategy he/she follows, the seat reviews should be deleted.

delete from Seat_Review

where Review_ID = '000003';

Before deletion on Seat Review Table

	client_id	event_id	venue_id	seat_id	review_id	seat_angle	seat_image	comment_date
78	14945614554	1214	1	000003	000003	5		2020-01-01
79	11985981251	1214	1	000004	000004	10		2020-01-01
80	18958959898	1214	1	000005	000005	0		2020-01-01
81	15620505060	1214	1	000006	000006	10		2020-01-01
82	15165195818	1214	1	000007	000007	0		2020-01-01
83	18951951581	1214	1	000008	000008	0		2020-01-01
84	19848948981	1214	1	000009	000009	0		2020-01-01
85	18098680134	1214	1	111111	000001	5		2020-01-01
86	10561898914	1676	12	000111	000111	5		2003-01-01
87	15814985519	1676	12	000112	000112	0		2003-01-01
88	16059159050	1676	12	000113	000113	5		2003-01-01
89	17818495050	1676	12	000114	000114	10		2003-01-01
90	14845919514	1676	12	000115	000115	0		2003-01-01
91	11591985191	1676	12	000116	000116	10		2003-01-01
92	12059591958	1676	12	000117	000117	0		2003-01-01

After deletion on Seat Review Table

	client_id	event_id	venue_id	seat_id	review_id	seat_angle	seat_image	comment_date
77	18915615959	1293	888	000078	000078	0		1948-01-01
78	11985981251	1214	1	000004	000004	10		2020-01-01
79	18958959898	1214	1	000005	000005	0		2020-01-01
80	15620505060	1214	1	000006	000006	10		2020-01-01
81	15165195818	1214	1	000007	000007	0		2020-01-01
82	18951951581	1214	1	000008	000008	0		2020-01-01
83	19848948981	1214	1	000009	000009	0		2020-01-01
84	18098680134	1214	1	111111	000001	5		2020-01-01
85	10561898914	1676	12	000111	000111	5		2003-01-01
86	15814985519	1676	12	000112	000112	0		2003-01-01
87	16059159050	1676	12	000113	000113	5		2003-01-01
88	17818495050	1676	12	000114	000114	10		2003-01-01
89	14845919514	1676	12	000115	000115	0		2003-01-01
90	11591985191	1676	12	000116	000116	10		2003-01-01
91	12059591958	1676	12	000117	000117	0		2003-01-01

Update 9: Update existing hour in Event Days Table.

update Event_days

set Starting_Hours = '12:00'

where Working_Hours = '20:30';

Before updating Event Days Table

	event_id	working_days	starting_hours	working_hours
6	1785	Wednesday-Thursday	13:00:00	3
7	1573	Wednesday-Friday	12:30:00	6
8	1025	Thursday-Friday	12:35:00	3
9	1783	Friday-Saturday	10:00:00	2
10	1023	Wednesday-Thursday	17:00:00	2
11	1092	Monday-Friday	20:30:00	3
12	1293	Thursday-Friday	10:30:00	4
13	1862	Monday-Wednesday	14:00:00	6
14	1795	Monday-Wednesday	12:30:00	5
15	9516	Monday-Wednesday	14:30:00	3
16	9846	Tuesday-Friday	12:30:00	3
17	1985	Tuesday-Friday	12:30:00	3
18	7961	Thursday-Friday	11:00:00	2.5
19	5515	Tuesday	11:30:00	3
20	9573	Tuesday-Thursday	17:00:00	3

After updating Event Days Table

	event_id	working_days	starting_hours	working_hours
8	1025	Thursday-Friday	12:35:00	3
9	1783	Friday-Saturday	10:00:00	2
10	1023	Wednesday-Thursday	17:00:00	2
11	1293	Thursday-Friday	10:30:00	4
12	1862	Monday-Wednesday	14:00:00	6
13	1795	Monday-Wednesday	12:30:00	5
14	9516	Monday-Wednesday	14:30:00	3
15	9846	Tuesday-Friday	12:30:00	3
16	1985	Tuesday-Friday	12:30:00	3
17	7961	Thursday-Friday	11:00:00	2.5
18	5515	Tuesday	11:30:00	3
19	9573	Tuesday-Thursday	17:00:00	3
20	1811	Monday-Wednesday	10:15:00	2
21	6486	Tuesday-Friday	10:45:00	2
22	1092	Monday-Friday	12:00:00	3

Delete 9: Delete all tuple in Event Days Table that contains weekend events due to prohibitions in pandemic. In this case, this event should be moved to another date which means start/end date on Event Table and working days on Event Days Table should be updated.

delete from Event_days

where Working_Days **like** '%Saturday%' **or** Working_Days **like** '%Sunday%';

	event_id	working_days	starting_hours	working_hours
1	1214	Monday-Wednesday	11:30:00	2
2	1224	Monday-Tuesday	11:00:00	3
3	1981	Monday-Tuesday	10:00:00	3
4	1676	Monday-Tuesday	17:00:00	3
5	1237	Tuesday-Thursday	12:00:00	4
6	1785	Wednesday-Thursday	13:00:00	3
7	1573	Wednesday-Friday	12:30:00	6
8	1025	Thursday-Friday	12:35:00	3
9	1783	Friday-Saturday	10:00:00	2
10	1023	Wednesday-Thursday	17:00:00	2
11	1293	Thursday-Friday	10:30:00	4
12	1862	Monday-Wednesday	14:00:00	6
13	1795	Monday-Wednesday	12:30:00	5
14	9516	Monday-Wednesday	14:30:00	3
15	9846	Tuesday-Friday	12:30:00	3

	event_id	working_days	starting_hours	working_hours
1	1214	Monday-Wednesday	11:30:00	2
2	1224	Monday-Tuesday	11:00:00	3
3	1981	Monday-Tuesday	10:00:00	3
4	1676	Monday-Tuesday	17:00:00	3
5	1237	Tuesday-Thursday	12:00:00	4
6	1785	Wednesday-Thursday	13:00:00	3
7	1573	Wednesday-Friday	12:30:00	6
8	1025	Thursday-Friday	12:35:00	3
9	1023	Wednesday-Thursday	17:00:00	2
10	1293	Thursday-Friday	10:30:00	4
11	1862	Monday-Wednesday	14:00:00	6
12	1795	Monday-Wednesday	12:30:00	5
13	9516	Monday-Wednesday	14:30:00	3
14	9846	Tuesday-Friday	12:30:00	3
15	1985	Tuesday-Friday	12:30:00	3

Since the online ticketing system includes one-week events, it will be sufficient to delete the working days including Saturday and Sunday.

Update 10: Attended_Clients must be updated in Attend Table if someone joins the event after it started.

update Attend

set Attended_Clients = **true**

where Client_ID = '14945614554' **and** Event_ID = '1214';

Before updating Attend Table

	client_id	event_id	attended_clients
1	18098680134	1214	[v]
2	14945614554	1214	[]
3	11985981251	1214	[v]
4	18958959898	1214	[v]
5	15620505060	1214	[v]
6	15165195818	1214	[v]
7	18951951581	1214	[v]
8	19848948981	1214	[v]
9	11651526516	1214	[v]
10	15020955900	1237	[v]
11	10020566506	1237	[v]
12	10561898914	1237	[v]
13	19859819785	1237	[v]
14	15814985519	1237	[v]
15	15620505060	1237	[v]

After updating Attend Table

	client_id	event_id	attended_clients
136	14945614554	1985	[v]
137	11985981251	1985	[v]
138	19859819785	1985	[]
139	19848948981	1985	[v]
140	19846516516	1985	[v]
141	11651526516	1985	[v]
142	15995095950	1985	[v]
143	14945614554	1214	[v]
144	15020955900	1985	[v]
145	17561590505	1985	[v]
146	18789195516	1985	[v]
147	16059159050	9573	[v]
148	17115961595	9573	[v]
149	17818495050	9573	[v]
150	14845919514	9573	[v]

Delete 10: Delete tuple in Attend Table.

delete from Attend

where Client_ID = '10020566506' **and** Event_ID = '1237';

Before deletion on Attend Table

	client_id	event_id	attended_clients
3	18958959898	1214	[v]
4	15620505060	1214	[v]
5	15165195818	1214	[v]
6	18951951581	1214	[v]
7	19848948981	1214	[v]
8	11651526516	1214	[v]
9	15020955900	1237	[v]
10	10020566506	1237	[v]
11	10561898914	1237	[v]
12	19859819785	1237	[v]
13	15814985519	1237	[v]
14	15620505060	1237	[v]
15	14065159815	1237	[v]
16	17561590505	1237	[v]
17	18789195516	1237	[v]

After deletion on Attend Table

	client_id	event_id	attended_clients
3	18958959898	1214	[v]
4	15620505060	1214	[v]
5	15165195818	1214	[v]
6	18951951581	1214	[v]
7	19848948981	1214	[v]
8	11651526516	1214	[v]
9	15020955900	1237	[v]
10	10561898914	1237	[v]
11	19859819785	1237	[v]
12	15814985519	1237	[v]
13	15620505060	1237	[v]
14	14065159815	1237	[v]
15	17561590505	1237	[v]
16	18789195516	1237	[v]
17	16059159050	1237	[v]

Update 11: Due to a minimum price policy of competition agency, updated minimum price as 17.5 in Buy/Cancel Table.

update Buy_Cancel

set Seat_Price = '17.5'

where Seat_Price <= '15'

Before updating Buy_Cancel Table

	seat_id	ticket_id	seat_price	bought_seats	cancelled_seats
1	000003	CGS-HAL1-B3	15	[]	[]
2	000004	CGS-HAL1-C1	15	[v]	[]
3	000005	CGS-HAL1-C2	15	[v]	[]
4	000006	CGS-HAL1-C3	15	[v]	[]
5	000007	CGS-HAL1-D1	15	[v]	[]
6	000008	CGS-HAL1-A1	35	[v]	[]
7	000009	CGS-HAL1-A2	35	[v]	[]
8	000011	CGS-HAL2-B1	15	[v]	[]
9	000012	CGS-HAL2-B2	15	[v]	[]
10	000013	CGS-HAL2-B3	15	[v]	[]
11	000014	CGS-HAL2-C1	15	[v]	[]
12	000015	CGS-HAL2-C2	15	[v]	[]
13	000016	CGS-HAL2-C3	15	[v]	[]
14	000017	CGS-HAL2-D1	15	[v]	[]
15	000018	CGS-HAL2-A1	35	[v]	[]

After updating Buy_Cancel Table

	seat_id	ticket_id	seat_price	bought_seats	cancelled_seats
70	000134	CAN-HAL1-C1	35	[v]	[]
71	000135	CAN-HAL1-C2	35	[v]	[]
72	000136	CAN-HAL1-C3	35	[v]	[]
73	000003	CGS-HAL1-B3	17.5	[]	[]
74	000004	CGS-HAL1-C1	17.5	[v]	[]
75	000005	CGS-HAL1-C2	17.5	[v]	[]
76	000006	CGS-HAL1-C3	17.5	[v]	[]
77	000007	CGS-HAL1-D1	17.5	[v]	[]
78	000011	CGS-HAL2-B1	17.5	[v]	[]
79	000012	CGS-HAL2-B2	17.5	[v]	[]
80	000013	CGS-HAL2-B3	17.5	[v]	[]
81	000014	CGS-HAL2-C1	17.5	[v]	[]
82	000015	CGS-HAL2-C2	17.5	[v]	[]
83	000016	CGS-HAL2-C3	17.5	[v]	[]
84	000017	CGS-HAL2-D1	17.5	[v]	[]

Delete 11: Delete tuple in **Buy/Cancel Table**. Due to the coronavirus measures, it has been set for all participants to sit with one seat apart. Those who have already bought their tickets can attend, but after the prohibitions, the tickets will be deleted with this query. (This part has not been runned on DBeaver because the clients already attended to the event. The figure is only for showing what will be deleted. This delete query will be applied after the prohibitions have started.)

delete from Buy_Cancel

where Ticket_ID **like** '%0' **or** Ticket_ID **like** '%2' **or** Ticket_ID **like** '%4' **or** Ticket_ID **like** '%6' **or** Ticket_ID **like** '%8';

Before deletion on Buy/Cancel Table

	seat_id	ticket_id	123 seat_price	bought_seats	cancelled_seats
29	000051	LGO-HAL1-B1	30	[v]	[]
30	000052	LGO-HAL1-B2	30	[v]	[]
31	000053	LGO-HAL1-B3	30	[v]	[]
32	000054	LGO-HAL1-C1	30	[v]	[]
33	000055	LGO-HAL1-C2	30	[v]	[]
34	000056	LGO-HAL1-C3	30	[v]	[]
35	000057	LGO-HAL1-D1	30	[v]	[]
36	000058	LGO-HAL1-A1	70	[v]	[]
37	000059	LGO-HAL1-A2	70	[v]	[]
38	000060	LGO-HAL1-A3	70	[v]	[]
39	000078	AGD-HAL1-A1	35	[v]	[]
40	000079	AGD-HAL1-A2	35	[v]	[]
41	000080	AGD-HAL1-A3	35	[v]	[]
42	000088	AGD-HAL2-A1	35	[v]	[]
43	000089	AGD-HAL2-A2	35	[v]	[]

After deletion on Buy/Cancel Table

	seat_id	ticket_id	123 seat_price	bought_seats	cancelled_seats
20	000051	LGO-HAL1-B1	30	[v]	[]
21	000053	LGO-HAL1-B3	30	[v]	[]
22	000054	LGO-HAL1-C1	30	[v]	[]
23	000056	LGO-HAL1-C3	30	[v]	[]
24	000057	LGO-HAL1-D1	30	[v]	[]
25	000058	LGO-HAL1-A1	70	[v]	[]
26	000060	LGO-HAL1-A3	70	[v]	[]
27	000078	AGD-HAL1-A1	35	[v]	[]
28	000080	AGD-HAL1-A3	35	[v]	[]
29	000088	AGD-HAL2-A1	35	[v]	[]
30	000090	AGD-HAL2-A3	35	[]	[]
31	000098	AGD-HAL3-A1	35	[v]	[]
32	000100	AGD-HAL3-A3	35	[v]	[]
33	000111	ARC-HAL2-B1	20	[v]	[]
34	000113	ARC-HAL2-B3	20	[v]	[]

Update 12: Update existing name in **Cast View Table** due to misunderstandings or clerical errors.

update Cast_View

set Name = 'Yevegny'

where Name = 'Yevgeny';

Before updating Cast View Table

	casteventid	name	surname	profession	email
17	1092	Keira	Knightley	Actor	keira.knightley@gmail.com
18	1795	Caston	Leroux	Singer	casleroux@gmail.com
19	9516	Koray	Karaca	Actor	kkaraca@gmail.com
20	9846	Mehmet Ali	Toklu	Actor	matoklu@gmail.com
21	1985	Al	Pacino	Actor	alpacino@gmail.com
22	7961	Antonio	Ghislanzoni	Composer	aghislanzoni@gmail.com
23	5515	Yevgeny	Grinko	Pianist	yevgrinko@gmail.com
24	9573	Leonardo	Dicaprio	Actor	leo.dicaprio@gmail.com
25	1811	Susan	Sarandon	Actor	ssarandon@gmail.com
26	1811	Geena	Davis	Actor	geena.davis@gmail.com
27	1811	Brad	Pitt	Actor	brad.pitt@gmail.com
28	6486	Bill	Murray	Actor	bill.murray@gmail.com
29	6486	Andie	MacDowell	Actor	andie.macdowell@gmail.com
30	6486	Chris	Elliott	Actor	chris.elliott@gmail.com
31	1862	Fazil	Say	Pianist	Pianist.Fazil.Say@gmail.com

After updating Cast View Table

	casteventid	name	surname	profession	email
17	1092	Keira	Knightley	Actor	keira.knightley@gmail.com
18	1795	Caston	Leroux	Singer	casleroux@gmail.com
19	9516	Koray	Karaca	Actor	kkaraca@gmail.com
20	9846	Mehmet Ali	Toklu	Actor	matoklu@gmail.com
21	1985	Al	Pacino	Actor	alpacino@gmail.com
22	7961	Antonio	Ghislanzoni	Composer	aghislanzoni@gmail.com
23	9573	Leonardo	Dicaprio	Actor	leo.dicaprio@gmail.com
24	1811	Susan	Sarandon	Actor	ssarandon@gmail.com
25	1811	Geena	Davis	Actor	geena.davis@gmail.com
26	1811	Brad	Pitt	Actor	brad.pitt@gmail.com
27	6486	Bill	Murray	Actor	bill.murray@gmail.com
28	6486	Andie	MacDowell	Actor	andie.macdowell@gmail.com
29	6486	Chris	Elliott	Actor	chris.elliott@gmail.com
30	1862	Fazil	Say	Pianist	Pianist.Fazil.Say@gmail.com
31	5515	Yevegny	Grinko	Pianist	yevgrinko@gmail.com

Delete 12: Delete tuple **Cast View Table**. If the person in the cast leaves the event, the actor, cook, singer etc. must be deleted in that event.

delete from Cast_View

where Email = 'nur.yazar@gmail.com';

Before deletion on Cast View Table

	ABC casteventid	ABC name	ABC surname	ABC profession	ABC email
1	1214	Bülent Emin	Yarar	Actor	bemin.yarar@gmail.com
2	1224	Selena	Gomez	Artist	selena.gomez@gmail.com
3	1981	Vivica	Genaux	Artist	miss.vivica@gmail.com
4	1676	Daniel	Ratcliffe	Actor	daniel.ratcliffe@gmail.com
5	1676	Emma	Watson	Actor	emma.watson@gmail.com
6	1676	Rupertz	Grint	Actor	rupertz.grint@gmail.com
7	1237	Nur	Yazar	Actor	nur.yazar@gmail.com
8	1237	Sinem	Şahin	Actor	sinem.sahin@gmail.com
9	1237	E. Çağrı	Turan	Actor	cagri.turan@gmail.com
10	1785	Daniel	Ratcliffe	Actor	daniel.ratcliffe@gmail.com
11	1785	Emma	Watson	Actor	emma.watson@gmail.com
12	1785	Rupertz	Grint	Actor	rupertz.grint@gmail.com
13	1573	Tarkan	Tevetoğlu	Singer	tarkan@gmail.com
14	1025	Mazda	Özdemir	Actor	m.ozdemir@gmail.com
15	1783	Orhan	Avcı	Chef and Violinist	orhan.avci@gmail.com

After deletion on Cast View Table

	ABC casteventid	ABC name	ABC surname	ABC profession	ABC email
1	1214	Bülent Emin	Yarar	Actor	bemin.yarar@gmail.com
2	1224	Selena	Gomez	Artist	selena.gomez@gmail.com
3	1981	Vivica	Genaux	Artist	miss.vivica@gmail.com
4	1676	Daniel	Ratcliffe	Actor	daniel.ratcliffe@gmail.com
5	1676	Emma	Watson	Actor	emma.watson@gmail.com
6	1676	Rupertz	Grint	Actor	rupertz.grint@gmail.com
7	1237	Sinem	Şahin	Actor	sinem.sahin@gmail.com
8	1237	E. Çağrı	Turan	Actor	cagri.turan@gmail.com
9	1785	Daniel	Ratcliffe	Actor	daniel.ratcliffe@gmail.com
10	1785	Emma	Watson	Actor	emma.watson@gmail.com
11	1785	Rupertz	Grint	Actor	rupertz.grint@gmail.com
12	1573	Tarkan	Tevetoğlu	Singer	tarkan@gmail.com
13	1025	Mazda	Özdemir	Actor	m.ozdemir@gmail.com
14	1783	Orhan	Avcı	Chef and Violinist	orhan.avci@gmail.com
15	1023	Carrie	Fisher	Actor	carrie.fisher@gmail.com

Alter Table: Add “Email” column to the **Client Table** for customers who have a foreign phone number. Aim is minimize phone costs. Therefore, they can be reach via emails.

alter table Client

add column Email varchar(100);

update Client

set Email = 'caleb.stone@gmail.com'

where Client_ID = '16841818177';

Before updating Client Table

	client_id	name	surname
1	14945614554	Mehmet Batuhan	Şenol
2	11985981251	Umut	Fırat
3	18958959898	Ege	Aslan
4	15165195818	Berra	Çam
5	18951951581	Yeşim	Mete
6	19848948981	Asena	Anık
7	19846516516	Hakan	Gürgün
8	11651526516	Miray	Erdemli
9	15995095950	Başak	Arslan
10	15020955900	Berkan	Güneş
11	10020566506	Elif	Işık
12	10561898914	Esra	Koçak
13	19859819785	Meryem	Demircan
14	15814985519	Neslihan	Erdoğan
15	15620505060	Pınar	Ertenoğlu

After updating Client Table-using Alter Table

	client_id	name	surname	email
21	18561591599	Göksu	Tümer	[NULL]
22	11591985191	Umay	Güven	[NULL]
23	12059591958	Tuna	Özdemir	[NULL]
24	17988948919	Asya	Aktaş	[NULL]
25	18098680134	Günsu	Koçyiğit	[NULL]
26	15165195818	Berra	Çam	berra.cam@ç
27	19846516516	Hakan	Gürgün	hakan.gurgun
28	17115961595	Kemal	Şener	kensen@gm
29	16059159050	Ali Adil	Yaşlı	aa.yasli@gm
30	18956159198	Albina	Gülerman	albina.gulerm
31	10259059005	Öykü	Davud	oyku.davud@
32	13473735750	Lisa	Manoban	lisaa.mano@
33	13068841891	Jisoo	Kim	jisoki@gmail
34	16841818154	Henry	Federick	hefederick@
35	16841818177	Caleb	Stone	caleb.stone@

Queries:queries.sql

To find the average salary of all professions, the following query is written. As a result of the query, the table in Figure 1.1 is obtained.

```
select profession, avg_salary_of_profession
from( select profession, avg(salary) as avg_salary_of_profession
      from castt
      group by profession) as derivedTable
where avg_salary_of_profession > '2000';
```

	profession	avg_salary_of_profession
1	Pianist	471,000
2	Composer	599,000
3	Chef and Violinist	7,800
4	Singer	616,500
5	Artist	4,505,000
6	Actor	156,543.75

Figure 1.1

To see the average salary found for all professions in the descending order, the following query is written. As a result of the query, the table in Figure 1.2 is obtained.

```
select profession, avg_salary_of_profession
from( select profession, avg(salary) as avg_salary_of_profession
      from castt
      group by profession) as derivedTable
order by avg_salary_of_profession desc;
```

	profession	avg_salary_of_profession
1	Artist	4,505,000
2	Singer	616,500
3	Composer	599,000
4	Pianist	471,000
5	Actor	156,543.75
6	Chef and Violinist	7,800

Figure 1.2

To be able to see the events that are available for an employee on Friday afternoon leave, the following query is written. As a result of the query, the table in Figure 1.3 is obtained.

```
select Name, Starting_Hours
from Event_days ed, Event e
where Starting_Hours > '12.00' and ed.Event_ID= e.Event_ID
intersect
select Name, Starting_Hours
from Event_days ed, Event e
where Working_Days like '%Friday%' and ed.Event_ID= e.Event_ID;
```

	name	starting_hours
1	Lüküs Hayat	12:30:00
2	Siyahlı Kadın	12:35:00
3	Tarkan Concert	12:30:00
4	The Irishman	12:30:00

Figure 1.3

It has been suggested that events that are likely to occur again must have more than 7 scores. In order to ensure a good sale, events with a high score should be considered to be included next week's events schedule. To find events with an average score of more than 7, the following query is written. As a result of the query, the table in Figure 1.4 is obtained.

```
select Name, avg_score
from (select e.Name, avg(Given_Score) as avg_score
      from Event_Review er, Event e
      group by e.Name, er.ReviewEventID, e.Event_ID
      having er.ReviewEventID=e.Event_ID) as derivedTable
where avg_score > some (select avg(Given_Score) as avg_score
                       from Event_Review er, Event e
                       group by er.ReviewEventID, e.Event_ID
                       having er.ReviewEventID=e.Event_ID and avg(Given_Score) = '7');
```

	name	avg_score
1	Star Wars	7.225
2	Harry Potter and the Deathly Hallows – Part 2	8.26
3	Inception	9.5
4	Revival Tour	9.8
5	Operadaki Hayalet	7.75
6	Harry Potter and the Deathly Hallows – Part 1	8.7777777778

Figure 1.4

Events with an average score of 1-5 are considered low scoring events. To find low-scoring events, the following query is written. As a result of the query, the table in Figure 1.5 is obtained.

```
select e.Name, avg(Given_Score) as avg_score
from Event_Review er, Event e
group by e.Name, er.ReviewEventID, e.Event_ID
having er.ReviewEventID=e.Event_ID and avg(Given_Score) between '1' and '5';
```

	name	avg_score
1	Lüküs Hayat	4
2	Siyahlı Kadın	1
3	Tarkan Concert	4.75
4	Truva Sonatı	2
5	Europa Galante	3.5
6	Eyvah Nadir	3.5

Figure 1.5

To find the event with the maximum seat price and the maximum seat price among all events, the following query is written. As a result of the query, the table in Figure 1.6 is obtained.

```
select distinct Name, max(seat_price) as Max_Price
from Event e, Buy_Cancel c, Seat s
group by e.Name,e.Event_ID, s.Event_ID, s.Seat_ID
having e.Event_ID = s.Event_ID and s.Seat_ID in (select Seat_ID
                                                from Buy_Cancel
                                                where Seat_Price = (select max(Seat_Price) from Buy_Cancel));
```

	name	max_price
1	Revival Tour	1,000

Figure 1.6

To find the number of actors who played in the movie Harry Potter and Deathly Hollows-Part1, the following query is written. As a result of the query, the table in Figure 1.7 is obtained.

```
select E.Name, count(castt.email) as cast_count
from Castt, Event E
group by E.Name, CastEventID, Event_ID
having CastEventID= Event_ID and Event_ID = '1676';
```

	name	cast_count
1	Harry Potter and the Deathly Hallows – Part 1	3

Figure 1.7

To find minimum and maximum seat prices and events with these prices, the following query is written. As a result of the query, the table in Figure 1.8 is obtained.

```

select Name, max(seat_price) as Max_Price, null as Min_Price
from Event e, Buy_Cancel c, Seat s
group by e.Name,e.Event_ID, s.Event_ID, s.Seat_ID
having e.Event_ID = s.Event_ID and s.Seat_ID in (select Seat_ID
                                                    from Buy_Cancel
                                                    where Seat_Price in (select
max(Seat_Price) from Buy_Cancel))

union

select Name, null as Max_Price ,min(seat_price) as Min_Price
from Event e, Buy_Cancel c, Seat s
group by e.Name, e.Event_ID, s.Event_ID, s.Seat_ID, c.seat_id
having e.Event_ID = s.Event_ID and s.Seat_ID = c.Seat_ID and c.Seat_Price in (select Seat_Price
                                                    from Buy_Cancel
                                                    group by Seat_Price
                                                    having Seat_Price in (select min(Seat_Price) from
Buy_Cancel));

```

	name	max_price	min_price
1	Aida	[NULL]	17.5
2	Çalığışu	[NULL]	17.5
3	Hamlet	[NULL]	17.5
4	Operadaki Hayalet	[NULL]	17.5
5	Revival Tour	1,000	[NULL]
6	Siyahlı Kadın	[NULL]	17.5
7	Turandot	[NULL]	17.5

Figure 1.8

To find the total number of seat of a building's all saloon/halls, the following query is witten. As a result of the query, the table in Figure 1.9 is obtained.

```
select v.Name, sum(v.number_of_seat)
from Venue v
where v.name like 'Cüneyt Gökçer Stage'
group by v.Name;
```

	name	sum
1	Cüneyt Gökçer Stage	30

Figure 1.9

To find the clients who didn't watched the "Harry Potter and the Deathly Hallows – Part 1" and "Harry Potter and the Deathly Hallows – Part 2" and suggest them to watch the movies with the specialized ads, the following query is witten. As a result of the query, the table in Figure 1.10 is obtained.

```
select c.Client_ID, c.Name
from Client as c
where c.Client_ID not in (select distinct c.Client_ID
                        from Client as c, Attend as a
                        where a.Attended_Clients = true and (a.Event_ID = '1785' or a.Event_ID = '1676')
                        and a.Client_ID = c.Client_ID)
order by c.Name asc;
```

	client_id	name
1	14065159815	Abbas
2	18956159198	Albina
3	19848948981	Asena
4	15165195818	Berra
5	16841818177	Caleb
6	10020566506	Elif
7	18561591599	Göksu
8	19846516516	Hakan
9	16518948912	Hasan
10	16841818154	Henry
11	13068841891	Jisoo
12	17115961595	Kemal
13	18915615959	Kemalcan
14	13473735750	Lisa
15	19859819785	Meryem
16	10259059005	Öykü
17	15620505060	Pınar
18	11985981251	Umut
19	18951951581	Yeşim

Figure 1.10

To find out how many reviews each seat has, the following query is written. As a result of the query, the table in Figure 1.11 is obtained.

```
select Seat_ID, Seat_Name,
      ( select count(*)
        from Seat_Review
        where Seat.Seat_ID= Seat_Review. Seat_ID )
      as num_reviews
from Seat;
```

	seat_id	seat_name	num_reviews
1	000003	B3	0
2	000004	C1	1
3	000005	C2	1
4	000006	C3	1
5	000007	D1	1
6	000008	A1	1
7	000009	A2	1
8	000011	B1	1
9	000012	B2	1
10	000013	B3	1
11	000014	C1	1
12	000015	C2	1
13	000016	C3	1
14	000017	D1	1
15	000018	A1	1

Figure 1.11

To list the people who attended in the movie Pride and Prejudice, the following query is written. As a result of the query, the table in Figure 1.12 is obtained.

```
select distinct C.Client_ID, C.Name, C.Surname
from Client as C
where not exists ( (select Event_ID
                   from Event
                   where Event.Name = 'Pride and Prejudice')
                 except
                 (select A.Event_ID
                  from Attend as A
                  where A.Client_ID = C.Client_ID and A.Attended_Client = true));
```

	client_id	name	surname
1	11591985191	Umay	Güven
2	11651526516	Miray	Erdemli
3	11985981251	Umut	Fırat
4	12059591958	Tuna	Özdemir
5	14065159815	Abbas	Kurt
6	17988948919	Asya	Aktaş
7	18561591599	Göksu	Tümer
8	18956159198	Albina	Gülberman
9	19848948981	Asena	Anık

Figure 1.12

To list seats that have the best angle (the angle that directly sees the stage or the screen), the following query is written. As a result of the query, the table in Figure 1.13 is obtained. Also, the angles of the seats are shown in Figure 2.1.

```

select S.Seat_ID, S.Seat_Name
from Seat S, Seat_Review R, Event E
where S.Seat_ID = R.Seat_ID and S.Event_ID= E.Event_ID and E.Name= 'Europa Galante' and R.Seat_Angle <= all (select Seat_Angle
from Seat_Review R, Seat S
where S.Seat_ID = R.Seat_ID
intersect
select Seat_Angle
from Seat_Review R, Event E, Seat S
where S.Seat_ID = R.Seat_ID and S.Event_ID= E.Event_ID and E.Name= 'Europa Galante');

```

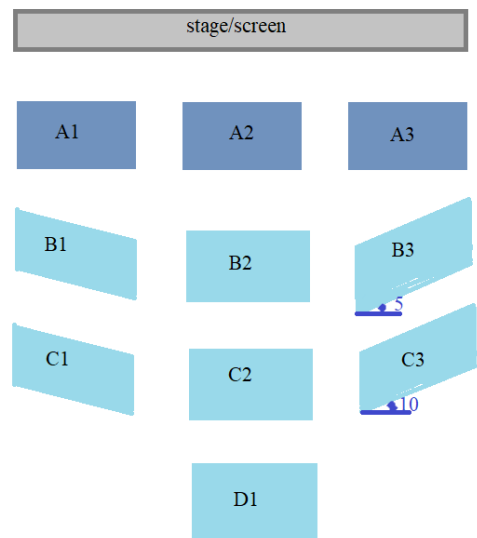


Figure 2.1

	ABS seat_id	ABS seat_name
1	000052	B2
2	000055	C2
3	000057	D1
4	000058	A1
5	000059	A2
6	000060	A3

Figure 1.13

To find the total salary of the players in the Çalığışu theater, the following query is written. As a result of the query, the table in Figure 1.14 is obtained.

```
select sum(salary)
from Castt join Event on Event_ID= CastEventID
where Event.Name = 'Çalığışu';
```

	sum
1	14,000

Figure 1.14

To list non-attend customers, the following query is written. As a result of the query, the table in Figure 1.15 is obtained.

```
select Client.Name, Event.Name
from Client natural join Attend, Event
where Attend.Event_ID = Event. Event_ID and Attend.Attended_Clients = false;
```

	name	name
1	Başak	Siyahlı Kadın
2	Kemalcan	Eyvah Nadir
3	Ege	Lüküs Hayat
4	Umut	Otto Barok
5	Esra	Operadaki Hayalet
6	Meryem	Operadaki Hayalet
7	Hasan	Operadaki Hayalet
8	Avşar	Operadaki Hayalet
9	Günsu	Aida
10	Avşar	Harry Potter and the Deathly Hallows – Part 2
11	Pınar	Harry Potter and the Deathly Hallows – Part 2
12	Berkan	Star Wars
13	Yeşim	Pride and Prejudice
14	Meryem	The Irishman
15	Hasan	Inception
16	Tuna	Inception
17	Başak	Tarkan Concert
18	Atakan	Tarkan Concert
19	Avşar	Truva Sonatı
20	Ege	Evgeny Grinko
21	Asena	Evgeny Grinko
22	Berkan	Evgeny Grinko

Figure 1.15

When the event and the venues they match are wanted to be seen as a cartesian product, if natural joining is used, the mismatched information will be lost. Left outer join is used to prevent information about the event from being lost. To see events without venues, the following query is written. As a result of the query, the table in Figure 1.16 is obtained.

select *

from Event e left outer join Venue v on e.Event_ID = v.VenueEventID;

	ABC event_id	ABC name	ABC genre	date_of_start	date_of_end	ABC venueeventid	ABC venue_id	ABC vphone_number	ABC name
1	1214	Hamlet	Theater	2020-03-23	2020-03-25	1214	1	3122400091	Cüneyt Gökçer Stage
2	1224	Revival Tour	Concert	2016-03-23	2020-03-24	1224	18	3122850385	Congresium Kongre ve Sergi Merkezi
3	1981	Europa Galante	Opera	2020-03-23	2020-03-24	1981	6	2125722070	Leyla Gencer Opera and Art Center
4	1676	Harry Potter and the Deathly Hallows – Part	Cinema	2020-03-23	2020-03-24	1676	12	3122411241	Arcadium Cinema
5	1237	Çalığışu	Theater	2020-03-24	2020-03-25	1237	2	3122400091	Cüneyt Gökçer Stage
6	1785	Harry Potter and the Deathly Hallows – Part	Cinema	2020-03-25	2020-05-26	1785	13	3122199393	Prestige Cinema
7	1573	Tarkan Concert	Concert	2020-03-25	2020-03-27	1573	19	2129123776	Zorlu Center Performing Arts Center
8	1025	Siyahlı Kadın	Theater	2020-03-26	2020-03-27	1025	3	3122400091	Cüneyt Gökçer Stage
9	1783	Otto Barok	Opera	2020-03-27	2020-03-28	[NULL]	[NULL]	[NULL]	[NULL]
10	1023	Star Wars	Cinema	2020-03-25	2020-03-26	1023	14	8502200967	Cinemaximum AnkaMall
11	1092	Pride and Prejudice	Cinema	2020-03-23	2020-03-27	1092	15	2124813383	Cinemaximum AnkaMall
12	1293	Turandot	Opera	2020-03-24	2020-03-27	1293	888	3123242214	Ankara General Directorate of Opera &
13	1862	Truva Sonatı	Concert	2020-03-23	2020-03-25	1862	20	8505490126	Jolly Joker
14	1795	Operadaki Hayalet	Opera	2020-03-23	2020-03-25	1795	9	3123242214	Ankara General Directorate of Opera &
15	9516	Eyvah Nadir	Theater	2020-03-23	2020-03-25	9516	4	3124671744	Şinasi Stage
16	9846	Lüküs Hayat	Theater	2020-03-24	2020-03-27	9846	5	3124671744	Şinasi Stage
17	1985	The Irishman	Cinema	2020-03-24	2020-03-27	1985	16	2124813383	Cinemaximum AnkaMall
18	7961	Aida	Opera	2020-03-26	2020-03-27	7961	10	3123242214	Ankara General Directorate of Opera &
19	5515	Evgeny Grinko	Concert	2020-03-24	2020-03-24	5515	21	2129123776	Babylon
20	9573	Inception	Cinema	2020-03-24	2020-03-26	9573	17	2124813383	Metropol Cinema
21	1811	Thelma and Louise	Cinema	2020-03-23	2020-03-25	[NULL]	[NULL]	[NULL]	[NULL]
22	6486	Groundhog Day	Cinema	2020-03-24	2020-03-27	[NULL]	[NULL]	[NULL]	[NULL]

Figure 1.16

To list clients/clients who have watched both Harry Potter movies, the following query is written. The full name of the Harry Potter movie is not remembered. Like all Harry Potter series, it is known that the name of the movie starts with Harry Potter and, this movie consists of two parts/sections/episodes. As a result of the query, the table in Figure 1.17 is obtained.

```

select C.Client_ID, C.Name, C.Surname
from Client as C
where not exists ( (select Event_ID
                    from Event
                    where Event.Name like 'Harry Potter%1'
                    except
                    select A.Event_ID
                    from Attend as A
                    where A.Client_ID = C.Client_ID and A.Attended_Clients = true)
union
(select Event_ID
 from Event
 where Event.Name like 'Harry Potter%2'
 except
 select A.Event_ID
 from Attend as A
 where A.Client_ID = C.Client_ID and A.Attended_Clients = true));

```

	client_id	name	surname
1	12059591958	Tuna	Özdemir

Figure 1.17

Special queries

To list the empty seats in the Hall 6 of the Metropol Cinema for each branch (Metropol Cinema has only one branch) in Turkey, the following query is written. As a result of the query, the table in Figure 2.1 is obtained.

```
select S.Seat_ID, S.Seat_Name as empty_seats
from Seat S, Buy_Cancel BC
where BC.Seat_ID = S.Seat_ID and (BC.Bought_seats = false or BC.Cancelled_seats = true)
intersect
select S.Seat_ID, S.Seat_Name as empty_seats
from Seat S, Venue V
where V.Venue_ID=S.Venue_ID and V.Section = 'Hall 6' and V.Name = 'Metropol Cinema'
```

	seat_id	empty_seats
1	000169	A2
2	000166	C3

Figure 2.1

To list the information of those who bought a VIP ticket for the Tarkan concert and did not attend the concert without canceling the ticket, the following query is written. As a result of the query, the table in Figure 2.2 is obtained.

```
select *
from Client C
where Client_ID=(select S.Client_ID
                from Seat S
                where S.Seat_Type='VIP'
                intersect
                select A.Client_ID
                from Attend A, Event e
                where A.Event_ID = (select E.Event_ID from Event e where E.Name='Tarkan Concert') and A.Attended_Clients=false);
```

	client_id	name	surname	email
1	17818495050	Atakan	Apan	[NULL]

Figure 2.2

Java Application Program

The java application of the database includes an user interface with 13 options. User selects the option they want to do (with entering a number between 0 and 12) and then, if required, application gets inputs from the user and executes the queries that user wanted to make.

Firstly, the external jar has been added to the java project and the main class of the project has been created. After that, the imports has been made and the connection, statement and resultset variables have been created. Local pc has been connected to the TEDU VPN and then the host, user and password informations have been filled.

Online Ticketing System Database

src

(default package)

Main.java

JRE System Library [JavaSE-14]

Referenced Libraries

postgresql-42.2.12.jre6.jar - C:\Users\fuatk\Downloads

```
1=import java.util.Scanner;
2 import java.sql.Connection;
3 import java.sql.Date;
4 import java.sql.DriverManager;
5 import java.sql.ResultSet;
6 import java.sql.ResultSetMetaData;
7 import java.sql.SQLException;
8 import java.sql.Statement;
9 import java.time.LocalDate;
10
11 public class Main {
12
13     private static Connection connection = null;
14     private static Statement statement = null;
15     private static ResultSet resultset = null;
16
17     final private static String host = "jdbc:postgresql://10.98.98.61:5432/group34";
18     final private static String user = "group34";
19     final private static String password = "d$b?XLuM*2*BmmK3";
20
```

The writeResultSet function has been created to print the datas in the database.

```
230 private static void writeResultSet(ResultSet resultSet)throws SQLException {
231     // TODO Auto-generated method stub
232     ResultSetMetaData rsmd = resultSet.getMetaData();
233     int columnsNumber = rsmd.getColumnCount();
234     while(resultSet.next()) {
235         for(int i=1; i<=columnsNumber; i++) {
236             System.out.print(resultSet.getString(i)+" ");
237         }
238         System.out.println();
239     }
240 }
```

The remote connection to the school server has been established and user interface has been created with the while loop.

```

27     try {
28         connection = DriverManager.getConnection(host, user, password);
29         System.out.println("Java JDBC PostgreSQL Example");
30
31         System.out.println("Connected to PostgreSQL database!");
32         statement = connection.createStatement();
33     }
34     catch(SQLException e){
35         System.out.println("Connection failure.");
36         e.printStackTrace();
37     }
38     System.out.println();
39
40     while(true) {
41         System.out.println("Online Ticketing System Database Menu");
42         System.out.println("Please select what you want to do:");
43         System.out.println("1-Insert a Client");
44         System.out.println("2-List the Clients");
45         System.out.println("3-Insert a Event");
46         System.out.println("4-List the Events");
47         System.out.println("5-Insert a Venue");
48         System.out.println("6-List the Venues in the selected city");
49
50         System.out.println("7-Update a Client's Surname");
51         System.out.println("8-Update a Event's ID");
52         System.out.println("9-Update a Venue's ID");
53
54         System.out.println("10-Delete a Client");
55         System.out.println("11-Delete a Event");
56         System.out.println("12-Delete a Venue");
57
58         System.out.println("0-Exit");
59     }

```

Then, with using if-else statements, the program has been adjusted to request the required inputs according to what they want to do (e.g requested the Event ID, Event Name, Event Genre, Event Date of Start/End for creating and inserting a new item; requested a City Name to list the Venues on that city) from the user and execute the required query (e.g insert into... for insert applications, select * from... for listing applications) for that application.

```

114     }
115     else if(input1 == 5) {
116         System.out.println("1-Enter the new Venue Event ID");
117         String newVenueEventID = keyboard.nextLine();
118         System.out.println("2-Enter the new Venue ID");
119         String newVenueID = keyboard.nextLine();
120         System.out.println("3-Enter the new Venue Phone Number");
121         String newPhoneNum = keyboard.nextLine();
122         System.out.println("4-Enter the new Venue Name");
123         String newVenueName = keyboard.nextLine();
124         System.out.println("5-Enter the new Venue Section");
125         String newVenueSec = keyboard.nextLine();
126         System.out.println("6-Enter the new Venue Information");
127         String newVenueInf = keyboard.nextLine();
128         System.out.println("7-Enter the new Venue's Number of Seat");
129         int newNumOfSeat = keyboard.nextInt();
130         keyboard.nextLine();
131         System.out.println("8-Enter the new Venue's State");
132         String newVenueState = keyboard.nextLine();
133         System.out.println("9-Enter the new Venue's City");
134         String newVenueCity = keyboard.nextLine();
135         System.out.println("10-Enter the new Venue's County");
136         String newVenueCounty = keyboard.nextLine();
137         System.out.println("11-Enter the new Venue's Street Name");
138         String newVenueStrName = keyboard.nextLine();
139         System.out.println("12-Enter the new Venue's Building No");
140         String newVenueBuildNo = keyboard.nextLine();
141
142         statement.executeUpdate("insert into Venue * (" + "venueeventid, venue_id, yphone_number, name, section, information, seating_arrangement, number_of_seat, state, city, county, street_name, building_no"
143         + "values (" + "newVenueEventID", " + "newVenueID", " + "newPhoneNum", " + "newVenueName", " + "newVenueSec", " + "newVenueInf", " + "null", " + "newNumOfSeat", " + "newVenueState", " + "newVenueCity", " + "newVenueCounty", " + "
144         System.out.println();
145     }
146
147     else if(input1 == 6) {
148         System.out.println("1-Enter the City");
149         String city = keyboard.nextLine();
150
151         resultSet = statement.executeQuery("select * from Venue where Venue.City = '" + city + "'");
152         writeResultSet(resultSet);
153         System.out.println();
154     }
155
156     else if(input1 == 7) {
157         System.out.println("1-Enter the surname of the client you want to update");
158         String oldSurname = keyboard.nextLine();
159         System.out.println("2-Enter the new surname of the client");
160         String newSurname = keyboard.nextLine();
161
162         statement.executeUpdate("update Client set Surname = '" + newSurname + "' where Surname = '" + oldSurname + "'");
163         System.out.println("The surname of the client updated as '" + newSurname + "'");
164         System.out.println();
165     }
166
167     else if(input1 == 8) {
168         System.out.println("1-Enter the ID of the event you want to update");
169         String oldID = keyboard.nextLine();
170         System.out.println("2-Enter the new ID of event");
171         String newID = keyboard.nextLine();
172
173         statement.executeUpdate("update Event set event_id = '" + newID + "' where event_id = '" + oldID + "'");
174         System.out.println("The event ID of the event updated to '" + newID + "'");
175         System.out.println();
176     }
177

```

Changes to Phase 1

- ❖ In the sql scripts of project section, the **Email attribute** has been added to the client relation by using the alter table. Since adding a field to the table is a design change, an email attribute has been added to all design stages.
- ❖ In venue relation, the Street **attribute** was changed to **County**, and the Street_Number **attribute** to **Building_No**. Both County and Building_No attributes have been changed in all design processes and in the create table section.
- ❖ **Seat_Name attribute** has been added to the seat relation. It has been added to all design processes and to the create table section.
- ❖ **Starting_Hours attribute** has been added to the event days relation. It has been added to all design processes and to the create table section.
- ❖ **Salary attribute** has been added to the cast relation. It has been added to all design processes and to the create table section.
- ❖ In event review relation, the Avg_Score **attribute** was changed to **Given_Score**. This attribute has been changed in all design processes and in the create table section.
- ❖ In the sql scripts of project section, **on delete cascade** and **on update cascade** have been added to the referential integrity constraints.
- ❖ The data **size** of some attributes has been **increased/decreased**. (e.g. Given_Score numeric(2,1) to (3,1), VPhone Number char (11) to char (10))