



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

Ακαδημαϊκό Έτος 2022-2023

Project – Εργαστηρίου Βάσεων Δεδομένων

«Travel Agency Management System»

Επώνυμο:	Μπακάλης	Μπάκνης	Σταμέλος
Όνομα:	Αθανάσιος	Γεώργιος	Χαρίλαος-Παναγιώτης
ΑΜ:	1084652	1084508	1090076
Email:	up1084652@upnet.gr	Up1084508@upnet.gr	up1090076@upnet.gr

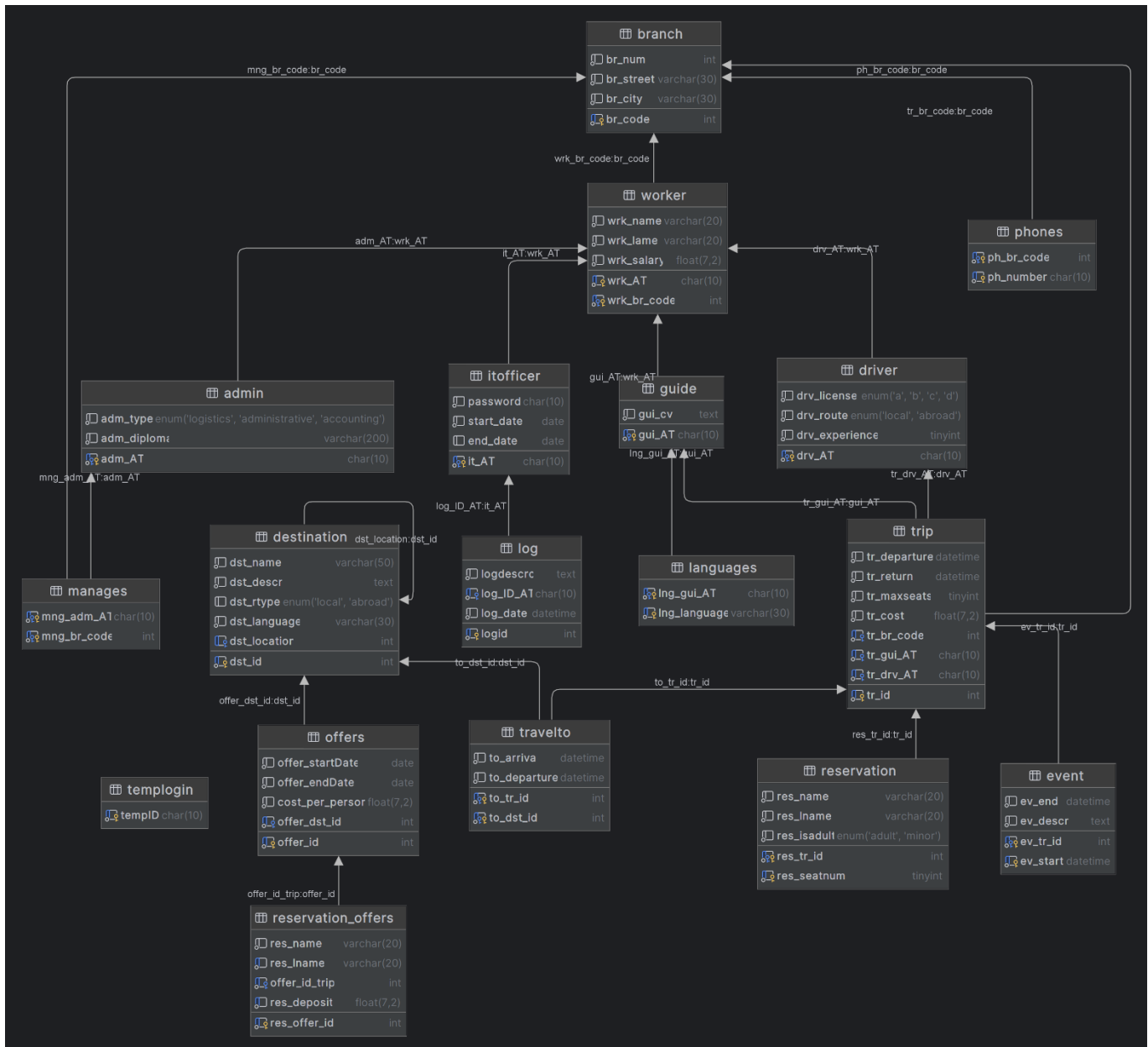
Περιεχόμενα:

Γενικά Σχόλια	4
1. Υλοποίηση Βάσης Δεδομένων	4
1.1 Σχεσιακό Διάγραμμα	4
2. Procedure Branch	15
2.1 Procedure1	15
2.2 Procedure2	16
2.3 Procedure3	17
2.4 Procedure4	19
2.4.1 Procedure 4.1	19
2.4.2 Procedure 4.2	20
3 Triggers	22
3.1 Triggers Information	22
3.2 Triggers 1	22
3.2.1 Trip triggers	22
3.2.2 Reservation Triggers	25
3.2.3 Event Triggers	28
3.2.4 travelTo Trigger	31
3.2.5 Destination Trigger	34
3.3 Triggers 2	38
3.4 Triggers 3	39
4 Υλοποίηση GUI	40
4.1 TravelAgency	40
4.2 ConnectionToDb	40
4.3 Login	40
4.4 Menu	43
4.5 Παραδείγματα	51
4.6 Procedures για έξτρα λειτουργίες	51
4.6 Procedures για έξτρα λειτουργίες	58
5 Μέθοδοι επιπλέον λειτουργικότητας	61
5.1 Επιπλέον μέθοδος λειτουργικότητας 1	62
5.2 Επιπλέον μέθοδος λειτουργικότητας 2	63
5.3 Επιπλέον procedures	64
6 Κώδικας MySQL	67
7 Κώδικας Java	92
7.1 ConnectionToDb	92
7.2 TravelAgency	92
7.3 Login	92
7.4 Menu	97

Γενικά Σχόλια

1. Υλοποίηση Βάσης Δεδομένων

1.1 Σχεσιακό Διάγραμμα



Στην παραπάνω εικόνα βλέπουμε το σχεσιακό διάγραμμα που ακολουθήσαμε για την υλοποίηση της βάσης δεδομένων. Παρατηρούμε όλους τους βασικούς πίνακες και τους πρόσθετους για την υλοποίηση του πρότζεκτ.

Βασικοί Πίνακες	Επιπλέον Πίνακες
branch	itOfficer
worker	log
phones	templogin
manages	reservation_offers
event	offers
trip	
destination	
travel_to	
reservation	
guide	
languages	
admin	
driver	

1. itOfficer

Αποθήκευση των workers που έχουν τον ρόλο του itOfficer.

2. log

Οι ενέργειες του Υπεύθυνου Πληροφορικής καταγράφονται στον πίνακα log.

3. Templogin

Αποθήκευση του τρέχων συνδεδεμένου χρήστη.

4. Reservation_offers

Κρατήσεις στις προσφορές ταξιδιών.

Κώδικας για τη δημιουργία των tables με inserts.

```
drop database if exists travelagency;
create database if not exists travelagency;
use travelagency;

create table if not exists branch(
  br_code int(11) not null auto_increment ,
  br_num int(4) not null,
  br_street varchar(30) default 'unknown' not null,
  br_city varchar(30) default 'unknown' not null,
  primary key(br_code)
);
```

```

create table if not exists phones(
    ph_br_code int(11) not null,
    ph_number char(10) default 'unknown' not null,
    primary key (ph_br_code,ph_number),
    constraint branchPhone foreign key (ph_br_code) references branch(br_code)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists destination(
    dst_id int(11) auto_increment,
    dst_name varchar(50) default 'unknown' not null,
    dst_descr text not null,
    dst_rtype enum('LOCAL','ABROAD'),
    dst_language varchar(30) default 'unknown' not null,
    dst_location int(11),
    primary key(dst_id),
    constraint locationId foreign key (dst_location) references destination(dst_id)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists worker(
    wrk_AT char(10) default 'unknown' not null,
    wrk_name varchar(20) default 'unknown' not null,
    wrk_lame varchar(20) default 'unknown' not null,
    wrk_salary float(7,2) not null,
    wrk_br_code int(11) not null ,
    PRIMARY KEY (wrk_AT, wrk_br_code), #Ένας εργαζόμενος εργάζεται σε ένα μόνο υποκατάστημα
    constraint workerBranch foreign key (wrk_br_code) references branch(br_code)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists guide(
    gui_AT char(10) not null,
    gui_cv text not null,
    PRIMARY KEY (gui_AT),
    constraint workerGuide foreign key (gui_AT) references worker(wrk_AT)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists languages(
    lng_gui_AT char(10) not null,
    lng_language varchar(30) default 'unknown' not null,
    primary key (lng_gui_AT,lng_language),
    constraint guideLanguage foreign key (lng_gui_AT) references guide(gui_AT)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists driver(
    drv_AT char(10) not null,
    drv_license enum('A','B','C','D') not null,
    drv_route enum('LOCAL','ABROAD') not null,
    drv_experience tinyint(4) not null,
    PRIMARY KEY(drv_AT),
    constraint driverWorkData foreign key (drv_AT) references worker(wrk_AT)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists admin(
    adm_AT char(10) not null,
    adm_type enum('LOGISTICS','ADMINISTRATIVE','ACCOUNTING') not null,
    adm_diploma varchar(200) DEFAULT 'unkown' not null,
    PRIMARY KEY(adm_AT),
    constraint workerAdminAt foreign key (adm_AT) references worker(wrk_AT)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists manages(
    mng_adm_AT char(10) default 'unkown' not null,
    mng_br_code int(11) not null ,

```

```

        primary key (mng_adm_AT, mng_br_code),
        constraint manageBranch foreign key (mng_br_code) references branch(br_code)
        ON DELETE CASCADE ON UPDATE CASCADE,
        constraint managerAdministration foreign key (mng_adm_AT) references admin(adm_AT)
        ON DELETE CASCADE ON UPDATE CASCADE
    );

create table if not exists trip(
    tr_id int(10) not null auto_increment,
    tr_departure datetime not null,
    tr_return datetime not null,
    tr_maxseats tinyint(4) not null,
    tr_cost float(7,2) not null,
    tr_br_code int(11) not null,
    tr_gui_AT char(10) not null,
    tr_drv_AT char(10) not null,
    PRIMARY KEY(tr_id),
    constraint tripBranchCode foreign key (tr_br_code) references branch(br_code)
    ON DELETE CASCADE ON UPDATE CASCADE,
    constraint tripGuideAt foreign key (tr_gui_AT) references guide(gui_AT)
    ON DELETE CASCADE ON UPDATE CASCADE,
    constraint tripDriverAt foreign key (tr_drv_AT) references driver(drv_AT)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists event(
    ev_tr_id int(11) not null,          #removed auto increment
    ev_start datetime not null,
    ev_end datetime not null,
    ev_descr text not null,
    primary key (ev_tr_id, ev_start),
    constraint tripEventId foreign key (ev_tr_id) references trip(tr_id)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists reservation(
    res_tr_id int(11) not null auto_increment,
    res_seatnum tinyint(4) not null,
    res_name varchar(20) default 'unknown' not null,
    res_lname varchar(20) default 'unknown' not null,
    res_isadult enum('ADULT','MINOR') not null,
    primary key (res_tr_id, res_seatnum),
    constraint tripReservationId foreign key (res_tr_id) references trip(tr_id)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists travelTo(
    to_tr_id int(11) not null auto_increment,
    to_dst_id int(11) not null,
    to_arrival datetime not null,
    to_departure datetime not null,
    primary key (to_tr_id, to_dst_id),
    constraint travelIdTrip foreign key (to_tr_id) references trip(tr_id)
    ON DELETE CASCADE ON UPDATE CASCADE,
    constraint destinationIdTrip foreign key (to_dst_id) references destination(dst_id)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists offers(
    offer_id int(11) not null auto_increment,
    offer_startDate date not null,
    offer_endDate date not null,
    cost_per_person float(7,2) not null,
    offer_dst_id int(11) not null,
    primary key (offer_id),
    constraint offerDestination foreign key (offer_dst_id) references destination(dst_id)
    on delete cascade on update cascade
);

create table if not exists reservation_offers(
    res_offer_id int(11) not null auto_increment,

```

```

    res_name varchar(20) not null,
    res_lname varchar(20) not null,
    offer_id_trip int(11) not null,
    res_deposit float(7,2) not null,
    primary key (res_offer_id),
    constraint offerId foreign key (offer_id_trip) references offers(offer_id)
    on delete cascade on update cascade
);

create table if not exists itOfficer(
    it_AT char(10) not null,
    password char(10) default 'password' not null,
    start_date date not null,
    end_date date,
    primary key (IT_AT),
    constraint itworker foreign key (IT_AT) references worker(wrk_AT)
    on delete cascade on update cascade
);

create table if not exists log(
    logid int not null auto_increment,
    logdescrc text not null,
    log_ID_AT char(10) not null,
    log_date datetime not null,
    primary key(logid),
    constraint logitat foreign key (log_ID_AT) references itOfficer(IT_AT)
    on delete cascade on update cascade
);

drop table if exists templogin;
create table templogin(
    tempID char(10) primary key
);

INSERT INTO branch VALUES
(1, 1, 'Main Street', 'New York'),
(2, 2, 'Maple Avenue', 'Chicago'),
(3, 3, '1st Street', 'Los Angeles'),
(4, 4, '2nd Avenue', 'Houston'),
(5, 5, '3rd Boulevard', 'Philadelphia'),
(6, 1, '4th Street', 'Phoenix'),
(7, 2, '5th Avenue', 'San Antonio'),
(8, 3, '6th Boulevard', 'San Diego'),
(9, 4, '7th Street', 'Dallas'),
(10, 5, '8th Avenue', 'San Jose');

INSERT INTO phones VALUES
(1, '0201111111'),
(2, '0202222222'),
(3, '0203333333'),
(4, '0204444444'),
(5, '0205555555'),
(6, '0206666666'),
(7, '0207777777'),
(8, '0208888888'),
(9, '0209999999'),
(10, '0200000000'),
(1, '0201234567'),
(2, '0202345678'),
(3, '0203456789'),
(4, '0204567890'),
(5, '0205678901'),
(6, '0206789012'),
(7, '0207890123'),
(8, '0208901234'),
(9, '0209012345'),
(10, '0200123456'),
(1, '0205432109'),
(2, '0204321098'),
(3, '0203210987'),

```



```

(4, '0202109876'),
(5, '0201098765'),
(6, '0200987654'),
(7, '0209876543'),
(8, '0208765432'),
(9, '0207654321'),
(10, '0206543210');

INSERT INTO destination VALUES
(1, 'New York', 'The city that never sleeps', 'LOCAL', 'English', NULL),
(2, 'Chicago', 'The Windy City', 'LOCAL', 'English', NULL),
(3, 'Los Angeles', 'The City of Angels', 'LOCAL', 'English', NULL),
(4, 'Houston', 'Space City', 'LOCAL', 'English', NULL),
(5, 'Philadelphia', 'The City of Brotherly Love', 'LOCAL', 'English', NULL),
(6, 'Phoenix', 'The Valley of the Sun', 'LOCAL', 'English', NULL),
(7, 'San Antonio', 'The Alamo City', 'LOCAL', 'English', NULL),
(8, 'San Diego', 'America's Finest City', 'LOCAL', 'English', NULL),
(9, 'Dallas', 'Big D', 'LOCAL', 'English', NULL),
(10, 'San Jose', 'The Capital of Silicon Valley', 'LOCAL', 'English', NULL),
(11, 'Athens', 'The Capital of Greece', 'LOCAL', 'Greek', 7),
(12, 'Thessaloniki', 'The Second City of Greece', 'LOCAL', 'Greek', 7),
(13, 'Rome', 'The Eternal City', 'ABROAD', 'Italian', 1),
(14, 'Paris', 'The City of Love', 'ABROAD', 'French', 2),
(15, 'London', 'The Capital of the UK', 'ABROAD', 'English', 3),
(16, 'Madrid', 'The Capital of Spain', 'ABROAD', 'Spanish', 4),
(17, 'Berlin', 'The Capital of Germany', 'ABROAD', 'German', 5),
(18, 'Tokyo', 'The Capital of Japan', 'ABROAD', 'Japanese', 6),
(19, 'Barcelona', 'The Capital of Catalonia', 'ABROAD', 'Spanish', NULL),
(20, 'Amsterdam', 'The Capital of the Netherlands', 'ABROAD', 'Dutch', NULL),
(21, 'Sydney', 'The Capital of Australia', 'ABROAD', 'English', NULL),
(22, 'Beijing', 'The Capital of China', 'ABROAD', 'Chinese', NULL),
(23, 'Moscow', 'The Capital of Russia', 'ABROAD', 'Russian', NULL),
(24, 'Dubai', 'The City of Gold', 'ABROAD', 'Arabic', NULL),
(25, 'Rio de Janeiro', 'The Marvelous City', 'ABROAD', 'Portuguese', NULL),
(26, 'Cape Town', 'The Mother City', 'ABROAD', 'English', NULL),
(27, 'Johannesburg', 'The City of Gold', 'ABROAD', 'Afrikaans', NULL),
(28, 'São Paulo', 'The Capital of São Paulo', 'ABROAD', 'Portuguese', NULL),
(29, 'Lisbon', 'The Capital of Portugal', 'ABROAD', 'Portuguese', NULL),
(30, 'Bangkok', 'The Capital of Thailand', 'ABROAD', 'Thai', NULL);

INSERT INTO worker VALUES
('AT000001', 'John', 'Doe', 45000.00, 1),
('AT000002', 'Jane', 'Doe', 50000.00, 2),
('AT000003', 'Bob', 'Smith', 40000.00, 3),
('AT000004', 'Sue', 'Smith', 55000.00, 4),
('AT000005', 'Alice', 'Jones', 52000.00, 5),
('AT000006', 'Mike', 'Jones', 48000.00, 6),
('AT000007', 'Sarah', 'Williams', 45000.00, 7),
('AT000008', 'Chris', 'Williams', 52000.00, 8),
('AT000009', 'Dave', 'Johnson', 50000.00, 9),
('AT000010', 'Emily', 'Johnson', 45000.00, 10),
('AT000011', 'Laura', 'Brown', 52000.00, 1),
('AT000012', 'Tom', 'Brown', 48000.00, 2),
('AT000013', 'Kim', 'Davis', 45000.00, 3),
('AT000014', 'Steve', 'Davis', 52000.00, 4),
('AT000015', 'Julie', 'Moore', 50000.00, 5),
('AT000016', 'Jason', 'Moore', 45000.00, 6),
('AT000017', 'Samantha', 'Miller', 52000.00, 7),
('AT000018', 'Mark', 'Miller', 48000.00, 8),
('AT000019', 'Alex', 'Wilson', 45000.00, 9),
('AT000020', 'Megan', 'Wilson', 52000.00, 10),
('AT000021', 'Tina', 'Taylor', 50000.00, 1),
('AT000022', 'Ryan', 'Taylor', 45000.00, 2),
('AT000023', 'Rachel', 'Anderson', 52000.00, 3),
('AT000024', 'Adam', 'Anderson', 48000.00, 4),
('AT000025', 'Katie', 'Thomas', 45000.00, 5),
('AT000026', 'Scott', 'Thomas', 52000.00, 6),
('AT000027', 'Amy', 'Jackson', 50000.00, 7),
('AT000028', 'Brian', 'Jackson', 45000.00, 8),
('AT000029', 'Lisa', 'White', 52000.00, 9),

```

```
(
    'AT000030', 'Matt', 'White', 48000.00, 10),
    ('AT000031', 'George', 'Baknis', 12000.00, 1),
    ('AT000032', 'Thanasis', 'Bakalis', 12000.00, 1),
    ('AT000033', 'Haris', 'Stamelos', 12000.00, 1),
    ('AT000034', 'Maria', 'Papadopoulou', 46000.00, 1),
    ('AT000035', 'Nikos', 'Georgiou', 48000.00, 2),
    ('AT000036', 'Giannis', 'Papadopoulos', 50000.00, 3),
    ('AT000037', 'Eleni', 'Katsarou', 45000.00, 4),
    ('AT000038', 'Dimitris', 'Nikolopoulos', 52000.00, 5),
    ('AT000039', 'Christina', 'Papadaki', 55000.00, 6),
    ('AT000040', 'Panos', 'Kostopoulos', 42000.00, 7),
    ('AT000041', 'Andreas', 'Ioannou', 48000.00, 8),
    ('AT000042', 'Sofia', 'Christou', 45000.00, 9),
    ('AT000043', 'Katerina', 'Papageorgiou', 52000.00, 10),
    ('AT000044', 'Marios', 'Antoniou', 48000.00, 1),
    ('AT000045', 'Stavros', 'Papadopoulos', 45000.00, 2),
    ('AT000046', 'Kostas', 'Vasilopoulos', 52000.00, 3),
    ('AT000047', 'Eva', 'Karakosta', 50000.00, 4),
    ('AT000048', 'Maria', 'Karagianni', 45000.00, 5),
    ('AT000049', 'Dimitris', 'Christopoulos', 52000.00, 6),
    ('AT000050', 'Eleni', 'Kontou', 48000.00, 7),
    ('AT000051', 'Sofia', 'Papaioannou', 45000.00, 8),
    ('AT000052', 'Katerina', 'Papadimitriou', 52000.00, 9),
    ('AT000053', 'Panos', 'Stefanopoulos', 50000.00, 10),
    ('AT000054', 'Stavros', 'Nikolaidis', 45000.00, 1),
    ('AT000055', 'Kostas', 'Christodoulou', 52000.00, 2),
    ('AT000056', 'Eva', 'Vlachou', 48000.00, 3),
    ('AT000057', 'Maria', 'Papandreou', 45000.00, 4),
    ('AT000058', 'Dimitris', 'Andreadis', 52000.00, 5),
    ('AT000059', 'Eleni', 'Mavridou', 48000.00, 6),
    ('AT000060', 'Sofia', 'Papadopoulou', 45000.00, 7);
```

```
INSERT INTO guide VALUES
```

```
(
    'AT000001', 'I have been working as a tour guide for 10 years and have extensive knowledge of
    local history and landmarks.'),
    ('AT000002', 'I have a degree in art history and have been leading museum and gallery tours for
    the past 5 years.'),
    ('AT000003', 'I am fluent in Spanish, French, and English and have been leading international
    tours for the past 7 years.'),
    ('AT000004', 'I have a background in geology and have been leading hiking and outdoor adventure
    tours for the past 3 years.'),
    ('AT000005', 'I have been leading food and wine tours in the local area for the past 8 years and
    have a deep understanding of regional cuisine and culinary traditions.'),
    ('AT000006', 'I have a degree in theatre and have been leading backstage and behind-the-scenes
    tours at local performance venues for the past 5 years.'),
    ('AT000007', 'I am an experienced wildlife photographer and have been leading photography tours
    in local nature reserves for the past 7 years.'),
    ('AT000008', 'I have a background in engineering and have been leading technology and innovation
    tours for the past 3 years.'),
    ('AT000009', 'I am an expert in local architecture and have been leading walking tours of
    historical neighborhoods for the past 8 years.'),
    ('AT000010', 'I have a degree in music and have been leading music and entertainment tours for
    the past 5 years.');
```

```
INSERT INTO languages VALUES
```

```
(
    'AT000001', 'Spanish'),
    ('AT000001', 'French'),
    ('AT000002', 'Greek'),
    ('AT000003', 'Italian'),
    ('AT000003', 'French'),
    ('AT000004', 'Spanish'),
    ('AT000005', 'German'),
    ('AT000006', 'French'),
    ('AT000007', 'Portuguese'),
    ('AT000008', 'Italian'),
    ('AT000009', 'Greek'),
    ('AT000010', 'French');
```

```
INSERT INTO driver VALUES
```

```
(
    'AT000011', 'B', 'LOCAL', 5),
```

```
( 'AT000012', 'C', 'LOCAL', 10),
( 'AT000013', 'D', 'LOCAL', 15),
( 'AT000014', 'A', 'ABROAD', 5),
( 'AT000015', 'B', 'ABROAD', 10),
( 'AT000016', 'C', 'ABROAD', 15),
( 'AT000017', 'D', 'ABROAD', 5),
( 'AT000018', 'A', 'LOCAL', 10),
( 'AT000019', 'B', 'LOCAL', 15),
( 'AT000020', 'C', 'ABROAD', 5);
```

INSERT INTO admin VALUES

```
( 'AT000021', 'ADMINISTRATIVE', 'Bachelor''s degree in Business Administration'),
( 'AT000022', 'ADMINISTRATIVE', 'Bachelor''s degree in Finance'),
( 'AT000023', 'ADMINISTRATIVE', 'Bachelor''s degree in Public Administration'),
( 'AT000024', 'ADMINISTRATIVE', 'Bachelor''s degree in Supply Chain Management'),
( 'AT000025', 'ADMINISTRATIVE', 'Bachelor''s degree in Accounting'),
( 'AT000026', 'ADMINISTRATIVE', 'Bachelor''s degree in Organizational Management'),
( 'AT000027', 'ADMINISTRATIVE', 'Bachelor''s degree in Logistics and Transportation'),
( 'AT000028', 'ADMINISTRATIVE', 'Bachelor''s degree in Financial Management'),
( 'AT000029', 'ADMINISTRATIVE', 'Bachelor''s degree in Business Administration'),
( 'AT000030', 'ADMINISTRATIVE', 'Bachelor''s degree in Industrial Management'),
( 'AT000031', 'ACCOUNTING', 'Bachelor''s degree in Health Services Administration'),
( 'AT000032', 'ACCOUNTING', 'Bachelor''s degree in Criminal Justice Administration'),
( 'AT000033', 'ACCOUNTING', 'Bachelor''s degree in Education Administration'),
( 'AT000034', 'ACCOUNTING', 'Bachelor''s degree in Hospitality Administration'),
( 'AT000035', 'ACCOUNTING', 'Bachelor''s degree in Environmental Policy and Management'),
( 'AT000036', 'ACCOUNTING', 'Bachelor''s degree in International Business'),
( 'AT000037', 'ACCOUNTING', 'Bachelor''s degree in Public Policy'),
( 'AT000038', 'ACCOUNTING', 'Bachelor''s degree in Project Management'),
( 'AT000039', 'ACCOUNTING', 'Bachelor''s degree in Social Work Administration'),
( 'AT000040', 'ACCOUNTING', 'Bachelor''s degree in Nonprofit Administration'),
( 'AT000041', 'LOGISTICS', 'Bachelor''s degree in Communication Studies'),
( 'AT000042', 'LOGISTICS', 'Bachelor''s degree in Information Technology Management'),
( 'AT000043', 'LOGISTICS', 'Bachelor''s degree in Marketing Administration'),
( 'AT000044', 'LOGISTICS', 'Bachelor''s degree in Retail Management'),
( 'AT000045', 'LOGISTICS', 'Bachelor''s degree in Sports Administration'),
( 'AT000046', 'LOGISTICS', 'Bachelor''s degree in Public Safety Administration'),
( 'AT000047', 'LOGISTICS', 'Bachelor''s degree in Emergency Management'),
( 'AT000048', 'LOGISTICS', 'Bachelor''s degree in Emergency Management'),
( 'AT000049', 'LOGISTICS', 'Bachelor''s degree in Emergency Management'),
( 'AT000050', 'LOGISTICS', 'Bachelor''s degree in Emergency Management');
```

INSERT INTO manages VALUES

```
( 'AT000021', 1),
( 'AT000022', 2),
( 'AT000023', 3),
( 'AT000024', 4),
( 'AT000025', 5),
( 'AT000026', 6),
( 'AT000027', 7),
( 'AT000028', 8),
( 'AT000029', 9),
( 'AT000030', 10),
( 'AT000031', 1),
( 'AT000032', 2),
( 'AT000033', 3),
( 'AT000034', 4),
( 'AT000035', 5),
( 'AT000036', 6),
( 'AT000037', 7),
( 'AT000038', 8),
( 'AT000039', 9),
( 'AT000040', 10),
( 'AT000041', 1),
( 'AT000042', 2),
( 'AT000043', 3),
( 'AT000044', 4),
( 'AT000045', 5),
( 'AT000046', 6),
```

```
('AT000047', 7),  
('AT000048', 8),  
('AT000049', 9),  
('AT000050', 10);
```

```
INSERT INTO trip VALUES
```

```
(1, '2022-01-01', '2022-01-07', 30, 1000.00, 1, 'AT000001', 'AT000011'),  
(2, '2022-01-08', '2022-01-14', 35, 1200.00, 2, 'AT000002', 'AT000012'),  
(3, '2022-01-15', '2022-01-21', 40, 1500.00, 3, 'AT000003', 'AT000013'),  
(4, '2022-01-22', '2022-01-28', 45, 1700.00, 4, 'AT000004', 'AT000014'),  
(5, '2022-01-29', '2022-02-04', 50, 2000.00, 5, 'AT000005', 'AT000015'),  
(6, '2022-02-05', '2022-02-11', 55, 2200.00, 6, 'AT000006', 'AT000016'),  
(7, '2022-02-12', '2022-02-18', 60, 2500.00, 7, 'AT000007', 'AT000017'),  
(8, '2022-02-19', '2022-02-25', 65, 2700.00, 8, 'AT000008', 'AT000018'),  
(9, '2022-02-26', '2022-03-04', 70, 3000.00, 9, 'AT000009', 'AT000019'),  
(10, '2022-03-05', '2022-03-11', 75, 3200.00, 10, 'AT000010', 'AT000020'),  
(11, '2022-03-12', '2022-03-18', 80, 3500.00, 1, 'AT000001', 'AT000011'),  
(12, '2022-03-19', '2022-03-25', 85, 3700.00, 2, 'AT000002', 'AT000012'),  
(13, '2022-03-26', '2022-04-01', 90, 4000.00, 3, 'AT000003', 'AT000013'),  
(14, '2022-04-02', '2022-04-08', 95, 4200.00, 4, 'AT000004', 'AT000014'),  
(15, '2022-04-09', '2022-04-15', 100, 4500.00, 5, 'AT000005', 'AT000015'),  
(16, '2022-04-16', '2022-04-22', 105, 4700.00, 6, 'AT000006', 'AT000016'),  
(17, '2022-04-23', '2022-04-29', 110, 5000.00, 7, 'AT000007', 'AT000017'),  
(18, '2022-04-30', '2022-05-06', 115, 5200.00, 8, 'AT000008', 'AT000018'),  
(19, '2022-05-07', '2022-05-13', 20, 1200.00, 9, 'AT000009', 'AT000019'),  
(20, '2022-05-14', '2022-05-20', 25, 1100.00, 10, 'AT000010', 'AT000020'),  
(21, '2022-05-21', '2022-05-27', 30, 1000.00, 1, 'AT000001', 'AT000011'),  
(22, '2022-05-28', '2022-06-03', 35, 900.00, 2, 'AT000002', 'AT000012'),  
(23, '2022-06-04', '2022-06-10', 40, 800.00, 3, 'AT000003', 'AT000013'),  
(24, '2022-06-11', '2022-06-17', 45, 700.00, 4, 'AT000004', 'AT000014'),  
(25, '2022-06-18', '2022-06-24', 50, 600.00, 5, 'AT000005', 'AT000015'),  
(26, '2022-06-25', '2022-07-01', 55, 500.00, 6, 'AT000006', 'AT000016'),  
(27, '2022-07-02', '2022-07-08', 60, 400.00, 7, 'AT000007', 'AT000017'),  
(28, '2022-07-09', '2022-07-15', 65, 300.00, 8, 'AT000008', 'AT000018'),  
(29, '2022-07-16', '2022-07-22', 70, 200.00, 9, 'AT000009', 'AT000019'),  
(30, '2022-07-23', '2022-07-29', 75, 100.00, 10, 'AT000010', 'AT000020');
```

```
INSERT INTO event VALUES
```

```
(1, '2022-01-01', '2022-01-07', 'Event 1'),  
(1, '2022-01-02', '2022-01-06', 'Event 1.1'),  
(2, '2022-01-08', '2022-01-14', 'Event 2'),  
(2, '2022-01-10', '2022-01-11', 'Event 2.2'),  
(3, '2022-01-15', '2022-01-21', 'Event 3'),  
(4, '2022-01-22', '2022-01-28', 'Event 4'),  
(5, '2022-01-29', '2022-02-04', 'Event 5'),  
(6, '2022-02-05', '2022-02-11', 'Event 6'),  
(7, '2022-02-12', '2022-02-18', 'Event 7'),  
(8, '2022-02-19', '2022-02-25', 'Event 8'),  
(9, '2022-02-26', '2022-03-04', 'Event 9'),  
(10, '2022-03-05', '2022-03-11', 'Event 10'),  
(11, '2022-03-12', '2022-03-18', 'Event 11'),  
(12, '2022-03-19', '2022-03-25', 'Event 12'),  
(13, '2022-03-26', '2022-04-01', 'Event 13'),  
(14, '2022-04-02', '2022-04-08', 'Event 14'),  
(15, '2022-04-09', '2022-04-15', 'Event 15'),  
(16, '2022-04-16', '2022-04-22', 'Event 16'),  
(17, '2022-04-23', '2022-04-29', 'Event 17'),  
(18, '2022-04-30', '2022-05-06', 'Event 18'),  
(19, '2022-05-07', '2022-05-13', 'Event 19'),  
(20, '2022-05-14', '2022-05-20', 'Event 20'),  
(21, '2022-05-21', '2022-05-27', 'Event 21'),  
(22, '2022-05-28', '2022-06-03', 'Event 22'),  
(23, '2022-06-04', '2022-06-10', 'Event 23'),  
(24, '2022-06-11', '2022-06-17', 'Event 24'),  
(25, '2022-06-18', '2022-06-24', 'Event 25'),  
(26, '2022-06-25', '2022-07-01', 'Event 26'),  
(27, '2022-07-02', '2022-07-08', 'Event 27'),  
(28, '2022-07-09', '2022-07-15', 'Event 28'),  
(29, '2022-07-16', '2022-07-22', 'Event 29'),  
(30, '2022-07-23', '2022-07-29', 'Event 30');
```

```

INSERT INTO reservation VALUES
(1, 1, 'John', 'Doe', 'ADULT'),
(1, 2, 'Jane', 'Doe', 'ADULT'),
(1, 3, 'Tom', 'Smith', 'MINOR'),
(2, 1, 'Alice', 'Johnson', 'ADULT'),
(2, 2, 'Bob', 'Johnson', 'MINOR'),
(3, 1, 'Emily', 'Williams', 'ADULT'),
(3, 2, 'William', 'Williams', 'ADULT'),
(4, 1, 'Samantha', 'Jones', 'ADULT'),
(4, 2, 'Jason', 'Jones', 'MINOR'),
(5, 1, 'Emma', 'Brown', 'ADULT'),
(5, 2, 'David', 'Brown', 'MINOR'),
(6, 1, 'Olivia', 'Davis', 'ADULT'),
(6, 2, 'James', 'Davis', 'MINOR'),
(7, 1, 'Ava', 'Miller', 'ADULT'),
(7, 2, 'Benjamin', 'Miller', 'MINOR'),
(8, 1, 'Isabella', 'Moore', 'ADULT'),
(8, 2, 'Michael', 'Moore', 'MINOR'),
(9, 1, 'Mia', 'Taylor', 'ADULT'),
(9, 2, 'Matthew', 'Taylor', 'MINOR'),
(10, 1, 'Abigail', 'Anderson', 'ADULT'),
(10, 2, 'Christopher', 'Anderson', 'MINOR'),
(11, 10, 'Emily', 'Thomas', 'ADULT'),
(1, 11, 'John', 'Doe', 'ADULT'),
(1, 21, 'Jane', 'Doe', 'ADULT'),
(1, 31, 'Bill', 'Smith', 'ADULT'),
(1, 41, 'Susan', 'Smith', 'ADULT'),
(1, 51, 'Paul', 'Johnson', 'MINOR'),
(2, 11, 'Emily', 'Johnson', 'MINOR'),
(2, 21, 'Michael', 'Williams', 'ADULT'),
(2, 31, 'Samantha', 'Williams', 'ADULT'),
(2, 41, 'Roger', 'Brown', 'ADULT');

INSERT INTO travelTo VALUES
(1, 1, '2022-01-07', '2022-01-01'),
(2, 2, '2022-01-14', '2022-01-08'),
(3, 3, '2022-01-21', '2022-01-15'),
(4, 4, '2022-01-28', '2022-01-22'),
(5, 5, '2022-02-04', '2022-01-29'),
(6, 6, '2022-02-11', '2022-02-05'),
(7, 7, '2022-02-18', '2022-02-12'),
(8, 8, '2022-02-25', '2022-02-19'),
(9, 9, '2022-03-04', '2022-02-26'),
(10, 10, '2022-03-11', '2022-03-05'),
(11, 11, '2022-03-18', '2022-03-12'),
(12, 12, '2022-03-25', '2022-03-19'),
(13, 13, '2022-04-01', '2022-03-26'),
(14, 14, '2022-04-08', '2022-04-02'),
(15, 15, '2022-04-15', '2022-04-09'),
(16, 16, '2022-04-22', '2022-04-16'),
(17, 17, '2022-04-29', '2022-04-23'),
(18, 18, '2022-05-06', '2022-04-30'),
(19, 19, '2022-05-13', '2022-05-07'),
(20, 20, '2022-05-20', '2022-05-14'),
(21, 21, '2022-05-27', '2022-05-21'),
(22, 22, '2022-06-03', '2022-05-28'),
(23, 23, '2022-06-10', '2022-06-04'),
(24, 24, '2022-06-17', '2022-06-11'),
(25, 25, '2022-06-24', '2022-06-18'),
(26, 26, '2022-07-01', '2022-06-25'),
(27, 27, '2022-07-08', '2022-07-02'),
(28, 28, '2022-07-15', '2022-07-09'),
(29, 29, '2022-07-22', '2022-07-16'),
(30, 30, '2022-07-29', '2022-07-23');

insert into itOfficer values('AT000051', '1234', NOW(), NULL);
insert into itOfficer values('AT000052', '1234', NOW(), NULL);
insert into itOfficer values('AT000053', '1234', NOW(), NULL);

```

```
insert into offers values
(null, '2023-01-01', '2024-01-01', 250, 1),
(null, '2023-01-10', '2024-02-01', 300, 2),
(null, '2023-05-01', '2024-05-25', 400, 3);
```

2. Procedure Branch

2.1 Procedure1

Στην πρώτη procedure με όνομα `newDriverAssignment` δίνεται ως είσοδος τα στοιχεία ενός νέου οδηγού και εντοπίζει το υποκατάστημα με τους λιγότερους οδηγούς και εισάγει τον οδηγό ως υπάλληλο του συγκεκριμένου υποκαταστήματος.

Κώδικας

```
#FIRST PROCEDURE
drop procedure if exists newDriverAssignment;
DELIMITER $

CREATE PROCEDURE newDriverAssignment(drvAT char(10), drvName varchar(20), drvLname varchar(30),
  drvSalary float(7,2), drvLicense enum('A', 'B', 'C', 'D'), drvRoute enum('LOCAL', 'ABROAD'),
  drvExperience tinyint(4))
BEGIN
  declare BRANCH_CODE int;

  select br_code into BRANCH_CODE
  from branch
  inner join worker on branch.br_code = worker.wrk_br_code
  inner join driver on worker.wrk_AT = driver.drv_AT
  group by br_code
  order by count(drv_AT) asc limit 1;

  insert into worker values(drvAT, drvName, drvLname, drvSalary, BRANCH_CODE);
  insert into driver values(drvAT, drvLicense, drvRoute, drvExperience);

END$

delimiter ;
```

```
call newDriverAssignment('AT000061' , 'George', 'Michalopoulos' , 12000.0 ,
'A' , 'ABROAD' , 10 );
```

Αποτελέσματα μετά την εκτέλεση της εντολής είναι η εισαγωγή του καινούιου driver στον πίνακα drivers.

```
select * from driver
```

	drv_AT	drv_license	drv_route	drv_experience
8	AT000018	A	LOCAL	10
9	AT000019	B	LOCAL	15
10	AT000020	C	ABROAD	5
11	AT000061	A	ABROAD	10

Εισαγωγή του driver στους workers

```
select * from worker
```

	wrk_AT	wrk_name	wrk_lame	wrk_salary	wrk_br_code
58	AT000058	Dimitris	Andreadis	52000	5
59	AT000059	Eleni	Mavridou	48000	6
60	AT000060	Sofia	Papadopoulou	45000	7
61	AT000061	George	Michalopoulos	12000	1

Ο driver / worker θα εισαχθεί στο υποκατάστημα με τους λιγότερους drivers στη συγκεκριμένη περίπτωση το branch με κωδικό 1.

2.2 Procedure2

Στη procedure με όνομα showAvailableTrips δίνεται ως όρισμα ο κωδικός ενός υποκαταστήματος και δύο ημερομηνίες. Για τα ταξίδια που διοργανώνονται από το υποκατάστημα του οποίου δόθηκε ο κωδικός, και των οποίων η ημερομηνία αναχώρησης είναι μέσα στο διάστημα που δόθηκε, θα επιστρέφονται ορισμένες πληροφορίες που φαίνονται παρακάτω.

```
#SECOND PROCEDURE
drop procedure if exists showAvailableTrips;
delimiter $

create procedure showAvailableTrips(brCode int(11), startDate date, endDate date)
begin
    declare not_found int;
    declare tripId int(11);
    declare tripCost float(7,2);
    declare maxSeats tinyint(4);
    declare reservCount tinyint(4);
    declare emptySeats tinyint(4);
    declare drivLname varchar(20);
    declare drivName varchar(20);
    declare GuideLname varchar(20);
    declare GuideName varchar(20);
    declare guideAT char(10);
    declare driverAt char(10);
    declare departureDate datetime;
    declare returnDate datetime;

    declare tcursor cursor for
    select tr_id, tr_departure, tr_return, tr_maxseats, tr_cost, tr_gui_AT, tr_drv_AT
    from trip
    where tr_br_code = brCode and tr_departure >= startDate and tr_departure <= endDate;

    declare continue handler for not found set not_found=1;

    set not_found=0;
    open tcursor;

    repeat
        fetch tcursor into tripId, departureDate, returnDate, maxSeats, tripCost, guideAT,
        driverAt;
        if(not_found=0) then
            select count(res_tr_id) into reservCount from reservation where res_tr_id = tripId;
            set emptySeats = maxSeats - reservCount;
            select wrk_lame, wrk_name into GuideLname, GuideName from worker where wrk_AT =
```



```

guideAT;
        select wrk_lame, wrk_name into drivLname, drivName from worker where wrk_AT =
driverAt;

        select tripCost, maxSeats, reservCount, emptySeats, drivLname,
                drivName, GuideLname, GuideName, departureDate, returnDate;
    else
        select null; #why?
    end if;
until (not_found=1)
end repeat;

close tcursor;
end $

DELIMITER ;

```

Παράδειγμα

```
call showAvailableTrips(1 , '2022-01-01', '2022-12-12');
```

	tripCost :	maxSeats :	reservCount :	emptySeats :	drivLname :	drivName :	GuideLname :	GuideName :	departureDate :	returnDate :
1	1000	30	8		22 Brown	Laura	Doe	John	2022-01-01 00:00:00	2022-01-07 00:00:00

	tripCost :	maxSeats :	reservCount :	emptySeats :	drivLname :	drivName :	GuideLname :	GuideName :	departureDate :	returnDate :
1	1000	30	0	30	Brown	Laura	Doe	John	2022-05-21 00:00:00	2022-05-27 00:00:00

	tripCost :	maxSeats :	reservCount :	emptySeats :	drivLname :	drivName :	GuideLname :	GuideName :	departureDate :	returnDate :
1	3500	80	1		79 Brown	Laura	Doe	John	2022-03-12 00:00:00	2022-03-18 00:00:00

2.3 Procedure3

Η stored procedure με όνομα `delete_admin` παίρνει ως όρισμα το όνομα και το επώνυμο ενός υπαλλήλου. Αν είναι διοικητικός τον διαγράφει. Αν ο υπάλληλος είναι διευθυντής υποκαταστήματος τότε εμφανίζεται μήνυμα ότι είναι διευθυντής του υποκαταστήματος και δεν επιτρέπει τη διαγραφή.

Κώδικας

```

#THIRD PROCEDURE
drop procedure if exists delete_admin;
DELIMITER $

CREATE PROCEDURE delete_admin(IN first_name VARCHAR(20), IN last_name VARCHAR(20))
BEGIN
    -- check if employee is a manager and his/her admin type.
    SELECT COUNT(*) INTO @is_manager
    FROM manages
    INNER JOIN worker ON manages.mng_adm_AT = worker.wrk_AT
    WHERE worker.wrk_name = first_name AND worker.wrk_lame = last_name;

    -- If the employee is a branch manager
    IF @is_manager > 0 THEN
        SELECT wrk_AT INTO @AT
        from worker
        INNER JOIN manages on manages.mng_adm_AT = wrk_AT
        WHERE worker.wrk_name = first_name AND worker.wrk_lame = last_name;

        -- And if he/she is of type ADMINISTRATIVE
        SELECT count(*) INTO @is_administrative
        from admin
        where admin.adm_AT = @AT AND admin.adm_type = 'ADMINISTRATIVE';
    
```

```

-- Cannot be deleted
IF @is_administrative > 0 THEN
    signal sqlstate value '45000'
    set message_text = 'Cannot delete administrator of a branch';

-- Else delete
ELSE
    DELETE admin FROM admin
    INNER JOIN worker ON admin.adm_AT = worker.wrk_AT
    WHERE worker.wrk_name = first_name AND worker.wrk_lame = last_name;
    select concat(first_name, ' ', last_name, ' ', 'with AT: ', @AT, 'deleted
successfully');
    #select 'not null';
    END IF;

-- Employee is not a manager
ELSE
    select concat(first_name, ' ', last_name, ' ', 'with AT: ', @AT, 'is not a manager. ');
    END IF;
END$

delimiter ;

```

Παράδειγμα

Προσπάθεια διαγραφής worker που είναι ADMINISTRATIVE

	wrk_AT	wrk_name	wrk_lame	wrk_salary	wrk_br_code
30	AT000030	Matt	White	48000	10

	adm_AT	adm_type	adm_diploma
10	AT000030	ADMINISTRATIVE	Bachelor's degree in Industrial Management

```
call delete_admin('Matt','White');
```

[45000][1644] Cannot delete administrator of a branch

Προσπάθεια διαγραφής worker που δεν είναι ADMINISTRATIVE

	wrk_AT	wrk_name	wrk_lame	wrk_salary	wrk_br_code
32	AT000032	Thanasis	Bakalis	12000	1

	adm_AT	adm_type	adm_diploma
12	AT000032	ACCOUNTING	Bachelor's degree in Criminal Justice Administration

```
call delete_admin('Thanasis','Bakalis');
```

	concat(first_name, ' ', last_name, ' ', 'with AT: ', @AT, 'deleted successfully')
1	Thanasis Bakalis with AT: AT000032deleted successfully

2.4 Procedure4

Η δημιουργία του index γίνεται μέσω της εντολής

```
CREATE INDEX idx ON reservation_offers (res_deposit, res_lname);
```

2.4.1 Procedure 4.1

Στη συγκεκριμένη procedure με όνομα `deposit_range_names` δίνονται ως παράμετροι εισόδου δύο τιμές (πχ 50, 200) και επιστρέφει τους πελάτες (επώνυμο, όνομα) που έκαναν κρατήσεις σε προσφορές ταξιδιών και πλήρωσαν για προκαταβολή, ποσό ανάμεσα στις δύο τιμές.

Κώδικας

```
#PROCEDURE 4.1
drop procedure if exists deposit_range_names;
delimiter $

create procedure deposit_range_names(IN minDeposit INT , IN maxDeposit INT)
BEGIN
    select res_name, res_lname from reservation_offers where res_deposit >= minDeposit and
res_deposit <= maxDeposit;
end $
delimiter ;

drop procedure if exists data_last_name;
delimiter $
```

Παράδειγμα

```
call deposit_range_names(50,200);
```

Με χρήση του index

```
travelagency> call deposit_range_names(50,200)
[2023-09-16 02:50:22] 500 rows retrieved starting from 1 in 198 ms (execution: 107 ms, fetching: 91 ms)
```

Χωρίς χρήση του index

```
travelagency> call deposit_range_names(50,200)
[2023-09-16 03:10:08] 500 rows retrieved starting from 1 in 188 ms (execution: 119 ms, fetching: 69 ms)
```

	res_name	res_lname
1	Itzel	Moyer
2	Thalia	Hernandez
3	Myla	Craig
4	Jazmine	Alvarez
5	Tristin	Banks
6	Aliyah	Holmes
7	Joselyn	Curtis
8	Harper	Walters
9	Augustus	Kent
10	Gaige	Compton
11	Tommy	Ewing
12	Denzel	Mathis
13	Karli	Vaughan
14	Milagros	Tate

2.4.2 Procedure 4.2

Η stored procedure με όνομα `data_last_name` στην οποία δίνεται ως παράμετρος εισόδου το επώνυμο ενός πελάτη και επιστρέφει τα ονόματα και επώνυμα των πελατών με το επώνυμο αυτό και η προσφορά ταξιδιού στην οποία έχει γίνει εγγραφή. Σε περίπτωση που υπάρχουν πάνω από ένας επιστρέφει το πλήθος των πελατών με το όνομα αυτό, ανά προσφορά ταξιδιού.

Κώδικας

```
drop procedure if exists data_last_name;
delimiter $

#PROCEDURE 4.2
create procedure data_last_name(IN lastName VARCHAR(20))
BEGIN
    select res_name, res_lname, offers.* from offers inner join reservation_offers on
    offers.offer_id = reservation_offers.offer_id_trip
    where reservation_offers.res_lname = lastName;

    if found_rows()>1 then
        select offers.*, count(res_lname) from offers
        inner join reservation_offers on offers.offer_id = reservation_offers.offer_id_trip
        where reservation_offers.res_lname = lastName
        group by offer_id;
    end if;
end $
delimiter ;
```

Παράδειγμα

```
call data_last_name( 'Moyer' );
```

	res_name	res_lname	offer_id	offer_startDate	offer_endDate	cost_per_person	offer_dst_id
1	Itzel	Moyer	1	2023-01-01	2024-01-01	250	1
2	Joel	Moyer	1	2023-01-01	2024-01-01	250	1
3	Cornelius	Moyer	1	2023-01-01	2024-01-01	250	1
4	Emmett	Moyer	1	2023-01-01	2024-01-01	250	1
5	Jayson	Moyer	1	2023-01-01	2024-01-01	250	1
6	Denise	Moyer	1	2023-01-01	2024-01-01	250	1
7	Damarion	Moyer	1	2023-01-01	2024-01-01	250	1
8	Libby	Moyer	1	2023-01-01	2024-01-01	250	1
9	Laila	Moyer	1	2023-01-01	2024-01-01	250	1
10	Christina	Moyer	1	2023-01-01	2024-01-01	250	1
11	Tyrone	Moyer	1	2023-01-01	2024-01-01	250	1
12	Itzel	Moyer	1	2023-01-01	2024-01-01	250	1
13	Joel	Moyer	1	2023-01-01	2024-01-01	250	1

	offer_id	offer_startDate	offer_endDate	cost_per_person	offer_dst_id	count(res_lname)
1	1	2023-01-01	2024-01-01	250	1	22
2	2	2023-01-10	2024-02-01	300	2	22
3	3	2023-05-01	2024-05-25	400	3	33

Με χρήση του index

```
travelagency> call data_last_name( 'Moyer' )
[2023-09-16 03:23:58] 3 rows retrieved starting from 1 in 282 ms (execution: 78 ms, fetching: 204 ms)
```

Χωρίς χρήση του index

```
travelagency> call data_last_name( 'Moyer' )
[2023-09-16 03:33:50] 3 rows retrieved starting from 1 in 197 ms (execution: 74 ms, fetching: 123 ms)
```

3 Triggers

3.1 Triggers Information

Για την υλοποίηση των triggers χρησιμοποιήθηκε η procedure getItID .

```
#procedure to get id for triggers
drop procedure if exists getItID;
delimiter $
create procedure getItID(in insertedItId char(10), in insertedItPass char(10), out itID char(10))
begin
    declare verificationValue char(10);

    select it_AT into verificationValue from itOfficer where it_AT = insertedItId and password =
insertedItPass;
    if (verificationValue is not null) then
        select insertedItId into itID;
        insert into templogin values(insertedItId);
    end if;
end $
```

3.2 Triggers 1

Trigger που θα ενημερώνουν το σχετικό πίνακα καταγραφής ενεργειών (log) για κάθε ενέργεια εισαγωγής, ενημέρωσης ή διαγραφής στους πίνακες trip, reservation, event, travel_to, destination, καθώς και το ποιος την εκτέλεσε.

3.2.1 Trip triggers

Κώδικας

```
drop trigger if exists tripInsertion;
delimiter $
create trigger tripInsertion after insert on trip
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Inserted a trip',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists tripUpdating;
```

```
delimiter $
create trigger tripUpdating after update on trip
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
            set logdescrc = 'Updated a trip.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists tripDeletion;
delimiter $
create trigger tripDeletion after delete on trip
for each row
begin

    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
            set logdescrc = 'Deleted a trip.',log_ID_AT=@loginID,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;
```

Παραδείγματα

INSERT

```
insert into trip values (31 , '2022-01-01', '2022-01-05' , 100 , 5000 , 1 , 'AT000001' , 'AT000011' );
```

	tr_id	tr_departure	tr_return	tr_maxseats	tr_cost	tr_br_code	tr_gui_AT	tr_drv_AT
8	8	2022-02-19 00:00:00	2022-02-25 00:00:00	65	2700	8	AT000008	AT000018
9	9	2022-02-26 00:00:00	2022-03-04 00:00:00	70	3000	9	AT000009	AT000019
10	10	2022-03-05 00:00:00	2022-03-11 00:00:00	75	3200	10	AT000010	AT000020
11	11	2022-03-12 00:00:00	2022-03-18 00:00:00	80	3500	1	AT000001	AT000011
12	12	2022-03-19 00:00:00	2022-03-25 00:00:00	85	3700	2	AT000002	AT000012
13	13	2022-03-26 00:00:00	2022-04-01 00:00:00	90	4000	3	AT000003	AT000013
14	14	2022-04-02 00:00:00	2022-04-08 00:00:00	95	4200	4	AT000004	AT000014
15	15	2022-04-09 00:00:00	2022-04-15 00:00:00	100	4500	5	AT000005	AT000015
16	16	2022-04-16 00:00:00	2022-04-22 00:00:00	105	4700	6	AT000006	AT000016
17	17	2022-04-23 00:00:00	2022-04-29 00:00:00	110	5000	7	AT000007	AT000017
18	18	2022-04-30 00:00:00	2022-05-06 00:00:00	115	5200	8	AT000008	AT000018
19	19	2022-05-07 00:00:00	2022-05-13 00:00:00	20	1200	9	AT000009	AT000019
20	20	2022-05-14 00:00:00	2022-05-20 00:00:00	25	1100	10	AT000010	AT000020
21	21	2022-05-21 00:00:00	2022-05-27 00:00:00	30	1000	1	AT000001	AT000011
22	22	2022-05-28 00:00:00	2022-06-03 00:00:00	35	900	2	AT000002	AT000012
23	23	2022-06-04 00:00:00	2022-06-10 00:00:00	40	800	3	AT000003	AT000013
24	24	2022-06-11 00:00:00	2022-06-17 00:00:00	45	700	4	AT000004	AT000014
25	25	2022-06-18 00:00:00	2022-06-24 00:00:00	50	600	5	AT000005	AT000015
26	26	2022-06-25 00:00:00	2022-07-01 00:00:00	55	500	6	AT000006	AT000016
27	27	2022-07-02 00:00:00	2022-07-08 00:00:00	60	400	7	AT000007	AT000017
28	28	2022-07-09 00:00:00	2022-07-15 00:00:00	65	300	8	AT000008	AT000018
29	29	2022-07-16 00:00:00	2022-07-22 00:00:00	70	200	9	AT000009	AT000019
30	30	2022-07-23 00:00:00	2022-07-29 00:00:00	75	100	10	AT000010	AT000020
31	31	2022-01-01 00:00:00	2022-01-05 00:00:00	100	5000	1	AT000001	AT000011

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip	AT000051	2023-09-16 14:01:52

UPDATE

```
update trip set tr_maxseats = 50 , tr_cost = 70000 where tr_id = 31 ;
```

	tr_id	tr_departure	tr_return	tr_maxseats	tr_cost	tr_br_code	tr_gui_AT	tr_drv_AT
31	31	2022-01-01 00:00:00	2022-01-05 00:00:00	50	70000	1	AT000001	AT000011

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10

DELETE

```
delete from trip where tr_id = 31;
```

	tr_id	tr_departure	tr_return	tr_maxseats	tr_cost	tr_br_code	tr_gui_AT	tr_drv_AT
7	7	2022-02-12 00:00:00	2022-02-18 00:00:00	60	2500	7	AT000007	AT000017
8	8	2022-02-19 00:00:00	2022-02-25 00:00:00	65	2700	8	AT000008	AT000018
9	9	2022-02-26 00:00:00	2022-03-04 00:00:00	70	3000	9	AT000009	AT000019
10	10	2022-03-05 00:00:00	2022-03-11 00:00:00	75	3200	10	AT000010	AT000020
11	11	2022-03-12 00:00:00	2022-03-18 00:00:00	80	3500	1	AT000001	AT000011
12	12	2022-03-19 00:00:00	2022-03-25 00:00:00	85	3700	2	AT000002	AT000012
13	13	2022-03-26 00:00:00	2022-04-01 00:00:00	90	4000	3	AT000003	AT000013
14	14	2022-04-02 00:00:00	2022-04-08 00:00:00	95	4200	4	AT000004	AT000014
15	15	2022-04-09 00:00:00	2022-04-15 00:00:00	100	4500	5	AT000005	AT000015
16	16	2022-04-16 00:00:00	2022-04-22 00:00:00	105	4700	6	AT000006	AT000016
17	17	2022-04-23 00:00:00	2022-04-29 00:00:00	110	5000	7	AT000007	AT000017
18	18	2022-04-30 00:00:00	2022-05-06 00:00:00	115	5200	8	AT000008	AT000018
19	19	2022-05-07 00:00:00	2022-05-13 00:00:00	20	1200	9	AT000009	AT000019
20	20	2022-05-14 00:00:00	2022-05-20 00:00:00	25	1100	10	AT000010	AT000020
21	21	2022-05-21 00:00:00	2022-05-27 00:00:00	30	1000	1	AT000001	AT000011
22	22	2022-05-28 00:00:00	2022-06-03 00:00:00	35	900	2	AT000002	AT000012
23	23	2022-06-04 00:00:00	2022-06-10 00:00:00	40	800	3	AT000003	AT000013
24	24	2022-06-11 00:00:00	2022-06-17 00:00:00	45	700	4	AT000004	AT000014
25	25	2022-06-18 00:00:00	2022-06-24 00:00:00	50	600	5	AT000005	AT000015
26	26	2022-06-25 00:00:00	2022-07-01 00:00:00	55	500	6	AT000006	AT000016
27	27	2022-07-02 00:00:00	2022-07-08 00:00:00	60	400	7	AT000007	AT000017
28	28	2022-07-09 00:00:00	2022-07-15 00:00:00	65	300	8	AT000008	AT000018
29	29	2022-07-16 00:00:00	2022-07-22 00:00:00	70	200	9	AT000009	AT000019
30	30	2022-07-23 00:00:00	2022-07-29 00:00:00	75	100	10	AT000010	AT000020

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53

3.2.2 Reservation Triggers

Κώδικας

```
drop trigger if exists reservationInsertion;
delimiter $
create trigger reservationInsertion after insert on reservation
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Inserted a reservation.', log_ID_AT=testvalue, log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;
```

```

drop trigger if exists reservationUpdating;
delimiter $
create trigger reservationUpdating after update on reservation
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Updated a reservation.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists reservationDeletion;
delimiter $
create trigger reservationDeletion after delete on reservation
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Deleted a reservation.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

```

Παραδείγματα

INSERT

```
insert into reservation values (30 , 1, 'Paul' , 'Johnson' , 'ADULT' );
```

	res_tr_id :	res_seatnum :	res_name :	res_lname :	res_isadult :
9	2	1	Alice	Johnson	ADULT
10	2	2	Bob	Johnson	MINOR
11	2	11	Emily	Johnson	MINOR
12	2	21	Michael	Williams	ADULT
13	2	31	Samantha	Williams	ADULT
14	2	41	Roger	Brown	ADULT
15	3	1	Emily	Williams	ADULT
16	3	2	William	Williams	ADULT
17	4	1	Samantha	Jones	ADULT
18	4	2	Jason	Jones	MINOR
19	5	1	Emma	Brown	ADULT
20	5	2	David	Brown	MINOR
21	6	1	Olivia	Davis	ADULT
22	6	2	James	Davis	MINOR
23	7	1	Ava	Miller	ADULT
24	7	2	Benjamin	Miller	MINOR
25	8	1	Isabella	Moore	ADULT
26	8	2	Michael	Moore	MINOR
27	9	1	Mia	Taylor	ADULT
28	9	2	Matthew	Taylor	MINOR
29	10	1	Abigail	Anderson	ADULT
30	10	2	Christopher	Anderson	MINOR
31	11	10	Emily	Thomas	ADULT
32	30	1	Paul	Johnson	ADULT

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53
4	4	Inserted a reserva...	AT000051	2023-09-16 14:29:05

UPDATE

```
update reservation set res_name = 'George' where res_tr_id = 30 and
res_seatnum = 1;
```

32	30	1	George	Johnson	ADULT
----	----	---	--------	---------	-------

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53
4	4	Inserted a reservation.	AT000051	2023-09-16 14:29:05
5	5	Updated a reservation.	AT000051	2023-09-16 14:30:25

DELETE

```
delete from reservation where res_tr_id = 30 and res_seatnum = 1 ;
```

	res_tr_id	res_seatnum	res_name	res_lname	res_isadult
8	1	51	Paul	Johnson	MINOR
9	2	1	Alice	Johnson	ADULT
10	2	2	Bob	Johnson	MINOR
11	2	11	Emily	Johnson	MINOR
12	2	21	Michael	Williams	ADULT
13	2	31	Samantha	Williams	ADULT
14	2	41	Roger	Brown	ADULT
15	3	1	Emily	Williams	ADULT
16	3	2	William	Williams	ADULT
17	4	1	Samantha	Jones	ADULT
18	4	2	Jason	Jones	MINOR
19	5	1	Emma	Brown	ADULT
20	5	2	David	Brown	MINOR
21	6	1	Olivia	Davis	ADULT
22	6	2	James	Davis	MINOR
23	7	1	Ava	Miller	ADULT
24	7	2	Benjamin	Miller	MINOR
25	8	1	Isabella	Moore	ADULT
26	8	2	Michael	Moore	MINOR
27	9	1	Mia	Taylor	ADULT
28	9	2	Matthew	Taylor	MINOR
29	10	1	Abigail	Anderson	ADULT
30	10	2	Christopher	Anderson	MINOR
31	11	10	Emily	Thomas	ADULT

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53
4	4	Inserted a reservation.	AT000051	2023-09-16 14:29:05
5	5	Updated a reservation.	AT000051	2023-09-16 14:30:25
6	6	Deleted a reservation.	AT000051	2023-09-16 14:31:34

3.2.3 Event Triggers

Κώδικας

```
drop trigger if exists eventInsertion;
delimiter $
create trigger eventInsertion after insert on event
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Inserted an event.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists eventUpdating;
delimiter $
create trigger eventUpdating after update on event
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Updated an event.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists eventDeletion;
delimiter $
create trigger eventDeletion after delete on event
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Deleted an event.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;
```

INSERT

```
insert into event values (30 , '2022-07-24' , '2022-07-25' , 'Event 31' );
```

	ev_tr_id	ev_start	ev_end	ev_descr
10	8	2022-02-19 00:00:00	2022-02-25 00:00:00	Event 8
11	9	2022-02-26 00:00:00	2022-03-04 00:00:00	Event 9
12	10	2022-03-05 00:00:00	2022-03-11 00:00:00	Event 10
13	11	2022-03-12 00:00:00	2022-03-18 00:00:00	Event 11
14	12	2022-03-19 00:00:00	2022-03-25 00:00:00	Event 12
15	13	2022-03-26 00:00:00	2022-04-01 00:00:00	Event 13
16	14	2022-04-02 00:00:00	2022-04-08 00:00:00	Event 14
17	15	2022-04-09 00:00:00	2022-04-15 00:00:00	Event 15
18	16	2022-04-16 00:00:00	2022-04-22 00:00:00	Event 16
19	17	2022-04-23 00:00:00	2022-04-29 00:00:00	Event 17
20	18	2022-04-30 00:00:00	2022-05-06 00:00:00	Event 18
21	19	2022-05-07 00:00:00	2022-05-13 00:00:00	Event 19
22	20	2022-05-14 00:00:00	2022-05-20 00:00:00	Event 20
23	21	2022-05-21 00:00:00	2022-05-27 00:00:00	Event 21
24	22	2022-05-28 00:00:00	2022-06-03 00:00:00	Event 22
25	23	2022-06-04 00:00:00	2022-06-10 00:00:00	Event 23
26	24	2022-06-11 00:00:00	2022-06-17 00:00:00	Event 24
27	25	2022-06-18 00:00:00	2022-06-24 00:00:00	Event 25
28	26	2022-06-25 00:00:00	2022-07-01 00:00:00	Event 26
29	27	2022-07-02 00:00:00	2022-07-08 00:00:00	Event 27
30	28	2022-07-09 00:00:00	2022-07-15 00:00:00	Event 28
31	29	2022-07-16 00:00:00	2022-07-22 00:00:00	Event 29
32	30	2022-07-23 00:00:00	2022-07-29 00:00:00	Event 30
33	30	2022-07-24 00:00:00	2022-07-25 00:00:00	Event 31

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53
4	4	Inserted a reservation.	AT000051	2023-09-16 14:29:05
5	5	Updated a reservation.	AT000051	2023-09-16 14:30:25
6	6	Deleted a reservation.	AT000051	2023-09-16 14:31:34
7	7	Inserted an event.	AT000051	2023-09-16 14:42:45
8	8	Deleted an event.	AT000051	2023-09-16 14:43:02
9	9	Inserted an event.	AT000051	2023-09-16 14:45:07

UPDATE

```
update event set ev_descr = 'The best event' where ev_tr_id = 30 and ev_start = '2022-07-24';
```

	ev_tr_id	ev_start	ev_end	ev_descr
10	8	2022-02-19 00:00:00	2022-02-25 00:00:00	Event 8
11	9	2022-02-26 00:00:00	2022-03-04 00:00:00	Event 9
12	10	2022-03-05 00:00:00	2022-03-11 00:00:00	Event 10
13	11	2022-03-12 00:00:00	2022-03-18 00:00:00	Event 11
14	12	2022-03-19 00:00:00	2022-03-25 00:00:00	Event 12
15	13	2022-03-26 00:00:00	2022-04-01 00:00:00	Event 13
16	14	2022-04-02 00:00:00	2022-04-08 00:00:00	Event 14
17	15	2022-04-09 00:00:00	2022-04-15 00:00:00	Event 15
18	16	2022-04-16 00:00:00	2022-04-22 00:00:00	Event 16
19	17	2022-04-23 00:00:00	2022-04-29 00:00:00	Event 17
20	18	2022-04-30 00:00:00	2022-05-06 00:00:00	Event 18
21	19	2022-05-07 00:00:00	2022-05-13 00:00:00	Event 19
22	20	2022-05-14 00:00:00	2022-05-20 00:00:00	Event 20
23	21	2022-05-21 00:00:00	2022-05-27 00:00:00	Event 21
24	22	2022-05-28 00:00:00	2022-06-03 00:00:00	Event 22
25	23	2022-06-04 00:00:00	2022-06-10 00:00:00	Event 23
26	24	2022-06-11 00:00:00	2022-06-17 00:00:00	Event 24
27	25	2022-06-18 00:00:00	2022-06-24 00:00:00	Event 25
28	26	2022-06-25 00:00:00	2022-07-01 00:00:00	Event 26
29	27	2022-07-02 00:00:00	2022-07-08 00:00:00	Event 27
30	28	2022-07-09 00:00:00	2022-07-15 00:00:00	Event 28
31	29	2022-07-16 00:00:00	2022-07-22 00:00:00	Event 29
32	30	2022-07-23 00:00:00	2022-07-29 00:00:00	Event 30
33	30	2022-07-24 00:00:00	2022-07-25 00:00:00	The best event

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip.	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53
4	4	Inserted a reservation.	AT000051	2023-09-16 14:29:05
5	5	Updated a reservation.	AT000051	2023-09-16 14:30:25
6	6	Deleted a reservation.	AT000051	2023-09-16 14:31:34
7	7	Inserted an event.	AT000051	2023-09-16 14:42:45
8	8	Deleted an event.	AT000051	2023-09-16 14:43:02
9	9	Inserted an event.	AT000051	2023-09-16 14:45:07
10	10	Updated an event.	AT000051	2023-09-16 14:46:59

DELETE

```
delete from event where ev_tr_id = 30 and ev_start = '2022-07-24';
```

	ev_tr_id	ev_start	ev_end	ev_descr
9	7	2022-02-12 00:00:00	2022-02-18 00:00:00	Event 7
10	8	2022-02-19 00:00:00	2022-02-25 00:00:00	Event 8
11	9	2022-02-26 00:00:00	2022-03-04 00:00:00	Event 9
12	10	2022-03-05 00:00:00	2022-03-11 00:00:00	Event 10
13	11	2022-03-12 00:00:00	2022-03-18 00:00:00	Event 11
14	12	2022-03-19 00:00:00	2022-03-25 00:00:00	Event 12
15	13	2022-03-26 00:00:00	2022-04-01 00:00:00	Event 13
16	14	2022-04-02 00:00:00	2022-04-08 00:00:00	Event 14
17	15	2022-04-09 00:00:00	2022-04-15 00:00:00	Event 15
18	16	2022-04-16 00:00:00	2022-04-22 00:00:00	Event 16
19	17	2022-04-23 00:00:00	2022-04-29 00:00:00	Event 17
20	18	2022-04-30 00:00:00	2022-05-06 00:00:00	Event 18
21	19	2022-05-07 00:00:00	2022-05-13 00:00:00	Event 19
22	20	2022-05-14 00:00:00	2022-05-20 00:00:00	Event 20
23	21	2022-05-21 00:00:00	2022-05-27 00:00:00	Event 21
24	22	2022-05-28 00:00:00	2022-06-03 00:00:00	Event 22
25	23	2022-06-04 00:00:00	2022-06-10 00:00:00	Event 23
26	24	2022-06-11 00:00:00	2022-06-17 00:00:00	Event 24
27	25	2022-06-18 00:00:00	2022-06-24 00:00:00	Event 25
28	26	2022-06-25 00:00:00	2022-07-01 00:00:00	Event 26
29	27	2022-07-02 00:00:00	2022-07-08 00:00:00	Event 27
30	28	2022-07-09 00:00:00	2022-07-15 00:00:00	Event 28
31	29	2022-07-16 00:00:00	2022-07-22 00:00:00	Event 29
32	30	2022-07-23 00:00:00	2022-07-29 00:00:00	Event 30

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip.	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53
4	4	Inserted a reservation.	AT000051	2023-09-16 14:29:05
5	5	Updated a reservation.	AT000051	2023-09-16 14:30:25
6	6	Deleted a reservation.	AT000051	2023-09-16 14:31:34
7	7	Inserted an event.	AT000051	2023-09-16 14:42:45
8	8	Deleted an event.	AT000051	2023-09-16 14:43:02
9	9	Inserted an event.	AT000051	2023-09-16 14:45:07
10	10	Updated an event.	AT000051	2023-09-16 14:46:59
11	11	Deleted an event.	AT000051	2023-09-16 14:48:04

3.2.4 travelTo Trigger

Κώδικας

```
drop trigger if exists traveltoUpdating;
delimiter $
create trigger traveltoUpdating after update on travelTo
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Updated a travelto.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists traveltoDeletion;
delimiter $
create trigger traveltoDeletion after delete on travelTo
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Deleted a travelto.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists destinationInsertion;
delimiter $
create trigger destinationInsertion after insert on destination
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Inserted a destination.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
```

```
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;
```

Παραδείγματα

DELETE

	to_tr_id	to_dst_id	to_arrival	to_departure
7	7	7	2022-02-18 00:00:00	2022-02-12 00:00:00
8	8	8	2022-02-25 00:00:00	2022-02-19 00:00:00
9	9	9	2022-03-04 00:00:00	2022-02-26 00:00:00
10	10	10	2022-03-11 00:00:00	2022-03-05 00:00:00
11	11	11	2022-03-18 00:00:00	2022-03-12 00:00:00
12	12	12	2022-03-25 00:00:00	2022-03-19 00:00:00
13	13	13	2022-04-01 00:00:00	2022-03-26 00:00:00
14	14	14	2022-04-08 00:00:00	2022-04-02 00:00:00
15	15	15	2022-04-15 00:00:00	2022-04-09 00:00:00
16	16	16	2022-04-22 00:00:00	2022-04-16 00:00:00
17	17	17	2022-04-29 00:00:00	2022-04-23 00:00:00
18	18	18	2022-05-06 00:00:00	2022-04-30 00:00:00
19	19	19	2022-05-13 00:00:00	2022-05-07 00:00:00
20	20	20	2022-05-20 00:00:00	2022-05-14 00:00:00
21	21	21	2022-05-27 00:00:00	2022-05-21 00:00:00
22	22	22	2022-06-03 00:00:00	2022-05-28 00:00:00
23	23	23	2022-06-10 00:00:00	2022-06-04 00:00:00
24	24	24	2022-06-17 00:00:00	2022-06-11 00:00:00
25	25	25	2022-06-24 00:00:00	2022-06-18 00:00:00
26	26	26	2022-07-01 00:00:00	2022-06-25 00:00:00
27	27	27	2022-07-08 00:00:00	2022-07-02 00:00:00
28	28	28	2022-07-15 00:00:00	2022-07-09 00:00:00
29	29	29	2022-07-22 00:00:00	2022-07-16 00:00:00
30	30	30	2022-07-29 00:00:00	2022-07-23 00:00:00

	to_tr_id	to_dst_id	to_arrival	to_departure
6	6	6	2022-02-11 00:00:00	2022-02-05 00:00:00
7	7	7	2022-02-18 00:00:00	2022-02-12 00:00:00
8	8	8	2022-02-25 00:00:00	2022-02-19 00:00:00
9	9	9	2022-03-04 00:00:00	2022-02-26 00:00:00
10	10	10	2022-03-11 00:00:00	2022-03-05 00:00:00
11	11	11	2022-03-18 00:00:00	2022-03-12 00:00:00
12	12	12	2022-03-25 00:00:00	2022-03-19 00:00:00
13	13	13	2022-04-01 00:00:00	2022-03-26 00:00:00
14	14	14	2022-04-08 00:00:00	2022-04-02 00:00:00
15	15	15	2022-04-15 00:00:00	2022-04-09 00:00:00
16	16	16	2022-04-22 00:00:00	2022-04-16 00:00:00
17	17	17	2022-04-29 00:00:00	2022-04-23 00:00:00
18	18	18	2022-05-06 00:00:00	2022-04-30 00:00:00
19	19	19	2022-05-13 00:00:00	2022-05-07 00:00:00
20	20	20	2022-05-20 00:00:00	2022-05-14 00:00:00
21	21	21	2022-05-27 00:00:00	2022-05-21 00:00:00
22	22	22	2022-06-03 00:00:00	2022-05-28 00:00:00
23	23	23	2022-06-10 00:00:00	2022-06-04 00:00:00
24	24	24	2022-06-17 00:00:00	2022-06-11 00:00:00
25	25	25	2022-06-24 00:00:00	2022-06-18 00:00:00
26	26	26	2022-07-01 00:00:00	2022-06-25 00:00:00
27	27	27	2022-07-08 00:00:00	2022-07-02 00:00:00
28	28	28	2022-07-15 00:00:00	2022-07-09 00:00:00
29	29	29	2022-07-22 00:00:00	2022-07-16 00:00:00

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53
4	4	Inserted a reservation.	AT000051	2023-09-16 14:29:05
5	5	Updated a reservation.	AT000051	2023-09-16 14:30:25
6	6	Deleted a reservation.	AT000051	2023-09-16 14:31:34
7	7	Inserted an event.	AT000051	2023-09-16 14:42:45
8	8	Deleted an event.	AT000051	2023-09-16 14:43:02
9	9	Inserted an event.	AT000051	2023-09-16 14:45:07
10	10	Updated an event.	AT000051	2023-09-16 14:46:59
11	11	Deleted an event.	AT000051	2023-09-16 14:48:04
12	12	Deleted a travelto.	AT000051	2023-09-16 15:05:31

INSERT

```
insert into travelTo values (30 , 30 , '2022-07-29', '2022-07-23');
```

	to_tr_id	to_dst_id	to_arrival	to_departure
6	6	6	2022-02-11 00:00:00	2022-02-05 00:00:00
7	7	7	2022-02-18 00:00:00	2022-02-12 00:00:00
8	8	8	2022-02-25 00:00:00	2022-02-19 00:00:00
9	9	9	2022-03-04 00:00:00	2022-02-26 00:00:00
10	10	10	2022-03-11 00:00:00	2022-03-05 00:00:00
11	11	11	2022-03-18 00:00:00	2022-03-12 00:00:00
12	12	12	2022-03-25 00:00:00	2022-03-19 00:00:00
13	13	13	2022-04-01 00:00:00	2022-03-26 00:00:00
14	14	14	2022-04-08 00:00:00	2022-04-02 00:00:00
15	15	15	2022-04-15 00:00:00	2022-04-09 00:00:00
16	16	16	2022-04-22 00:00:00	2022-04-16 00:00:00
17	17	17	2022-04-29 00:00:00	2022-04-23 00:00:00
18	18	18	2022-05-06 00:00:00	2022-04-30 00:00:00
19	19	19	2022-05-13 00:00:00	2022-05-07 00:00:00
20	20	20	2022-05-20 00:00:00	2022-05-14 00:00:00
21	21	21	2022-05-27 00:00:00	2022-05-21 00:00:00
22	22	22	2022-06-03 00:00:00	2022-05-28 00:00:00
23	23	23	2022-06-10 00:00:00	2022-06-04 00:00:00
24	24	24	2022-06-17 00:00:00	2022-06-11 00:00:00
25	25	25	2022-06-24 00:00:00	2022-06-18 00:00:00
26	26	26	2022-07-01 00:00:00	2022-06-25 00:00:00
27	27	27	2022-07-08 00:00:00	2022-07-02 00:00:00
28	28	28	2022-07-15 00:00:00	2022-07-09 00:00:00
29	29	29	2022-07-22 00:00:00	2022-07-16 00:00:00
30	30	30	2022-07-29 00:00:00	2022-07-23 00:00:00

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip.	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53
4	4	Inserted a reservation.	AT000051	2023-09-16 14:29:05
5	5	Updated a reservation.	AT000051	2023-09-16 14:30:25
6	6	Deleted a reservation.	AT000051	2023-09-16 14:31:34
7	7	Inserted an event.	AT000051	2023-09-16 14:42:45
8	8	Deleted an event.	AT000051	2023-09-16 14:43:02
9	9	Inserted an event.	AT000051	2023-09-16 14:45:07
10	10	Updated an event.	AT000051	2023-09-16 14:46:59
11	11	Deleted an event.	AT000051	2023-09-16 14:48:04
12	12	Deleted a travelto.	AT000051	2023-09-16 15:05:31
13	13	Inserted a travelto.	AT000051	2023-09-16 15:08:28

UPDATE

```
update travelTo set to_arrival = '2022-07-30' where to_tr_id = 30 and  
to_dst_id = 30;
```

	to_tr_id	to_dst_id	to_arrival	to_departure
6	6	6	2022-02-11 00:00:00	2022-02-05 00:00:00
7	7	7	2022-02-18 00:00:00	2022-02-12 00:00:00
8	8	8	2022-02-25 00:00:00	2022-02-19 00:00:00
9	9	9	2022-03-04 00:00:00	2022-02-26 00:00:00
10	10	10	2022-03-11 00:00:00	2022-03-05 00:00:00
11	11	11	2022-03-18 00:00:00	2022-03-12 00:00:00
12	12	12	2022-03-25 00:00:00	2022-03-19 00:00:00
13	13	13	2022-04-01 00:00:00	2022-03-26 00:00:00
14	14	14	2022-04-08 00:00:00	2022-04-02 00:00:00
15	15	15	2022-04-15 00:00:00	2022-04-09 00:00:00
16	16	16	2022-04-22 00:00:00	2022-04-16 00:00:00
17	17	17	2022-04-29 00:00:00	2022-04-23 00:00:00
18	18	18	2022-05-06 00:00:00	2022-04-30 00:00:00
19	19	19	2022-05-13 00:00:00	2022-05-07 00:00:00
20	20	20	2022-05-20 00:00:00	2022-05-14 00:00:00
21	21	21	2022-05-27 00:00:00	2022-05-21 00:00:00
22	22	22	2022-06-03 00:00:00	2022-05-28 00:00:00
23	23	23	2022-06-10 00:00:00	2022-06-04 00:00:00
24	24	24	2022-06-17 00:00:00	2022-06-11 00:00:00
25	25	25	2022-06-24 00:00:00	2022-06-18 00:00:00
26	26	26	2022-07-01 00:00:00	2022-06-25 00:00:00
27	27	27	2022-07-08 00:00:00	2022-07-02 00:00:00
28	28	28	2022-07-15 00:00:00	2022-07-09 00:00:00
29	29	29	2022-07-22 00:00:00	2022-07-16 00:00:00
30	30	30	2022-07-30 00:00:00	2022-07-23 00:00:00

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip.	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53
4	4	Inserted a reservation.	AT000051	2023-09-16 14:29:05
5	5	Updated a reservation.	AT000051	2023-09-16 14:30:25
6	6	Deleted a reservation.	AT000051	2023-09-16 14:31:34
7	7	Inserted an event.	AT000051	2023-09-16 14:42:45
8	8	Deleted an event.	AT000051	2023-09-16 14:43:02
9	9	Inserted an event.	AT000051	2023-09-16 14:45:07
10	10	Updated an event.	AT000051	2023-09-16 14:46:59
11	11	Deleted an event.	AT000051	2023-09-16 14:48:04
12	12	Deleted a travelto.	AT000051	2023-09-16 15:05:31
13	13	Inserted a travelto.	AT000051	2023-09-16 15:08:28
14	14	Updated a travelto.	AT000051	2023-09-16 15:13:05

3.2.5 Destination Trigger

Κώδικας

```
drop trigger if exists traveltoDeletion;
delimiter $
create trigger traveltoDeletion after delete on travelTo
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
```

```

        if(testvalue is not null) then
            insert into log
            set logdescrc = 'Deleted a travelto.',log_ID_AT=testvalue,log_date=now();
        else
            signal sqlstate value '45000'
            set message_text = 'Error!You have to login as It Officer.';
        end if;

end $
delimiter ;

drop trigger if exists destinationInsertion;
delimiter $
create trigger destinationInsertion after insert on destination
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Inserted a destination.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists destinationUpdating;
delimiter $
create trigger destinationUpdating after update on destination
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Updated a destination.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists destinationDeletion;
delimiter $
create trigger destinationDeletion after delete on destination
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Deleted a destination.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

```

INSERT

```

insert into destination values (31, 'China', 'Dragons country', 'ABROAD',
'Chinese', NULL);

```

	dst_id	dst_name	dst_descr	dst_rtype	dst_language	dst_location
31	31	China	Dragons country	ABROAD	Chinese	<null>

	logid	logdescri	log_ID_AT	log_date
1	1	Inserted a trip.	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53
4	4	Inserted a reservation.	AT000051	2023-09-16 14:29:05
5	5	Updated a reservation.	AT000051	2023-09-16 14:30:25
6	6	Deleted a reservation.	AT000051	2023-09-16 14:31:34
7	7	Inserted an event.	AT000051	2023-09-16 14:42:45
8	8	Deleted an event.	AT000051	2023-09-16 14:43:02
9	9	Inserted an event.	AT000051	2023-09-16 14:45:07
10	10	Updated an event.	AT000051	2023-09-16 14:46:59
11	11	Deleted an event.	AT000051	2023-09-16 14:48:04
12	12	Deleted a travelto.	AT000051	2023-09-16 15:05:31
13	13	Inserted a travelto.	AT000051	2023-09-16 15:08:28
14	14	Updated a travelto.	AT000051	2023-09-16 15:13:05
15	15	Inserted a destination.	AT000051	2023-09-16 15:25:14

UPDATE

```
update destination set dst_descr = 'Great wall' where dst_id = 31;
```

	dst_id	dst_name	dst_descr	dst_rtype	dst_language	dst_location
8	8	San Diego	America's Finest City	LOCAL	English	<null>
9	9	Dallas	Big D	LOCAL	English	<null>
10	10	San Jose	The Capital of Silicon Valley	LOCAL	English	<null>
11	11	Athens	The Capital of Greece	LOCAL	Greek	7
12	12	Thessaloniki	The Second City of Greece	LOCAL	Greek	7
13	13	Rome	The Eternal City	ABROAD	Italian	1
14	14	Paris	The City of Love	ABROAD	French	2
15	15	London	The Capital of the UK	ABROAD	English	3
16	16	Madrid	The Capital of Spain	ABROAD	Spanish	4
17	17	Berlin	The Capital of Germany	ABROAD	German	5
18	18	Tokyo	The Capital of Japan	ABROAD	Japanese	6
19	19	Barcelona	The Capital of Catalonia	ABROAD	Spanish	<null>
20	20	Amsterdam	The Capital of the Netherlands	ABROAD	Dutch	<null>
21	21	Sydney	The Capital of Australia	ABROAD	English	<null>
22	22	Beijing	The Capital of China	ABROAD	Chinese	<null>
23	23	Moscow	The Capital of Russia	ABROAD	Russian	<null>
24	24	Dubai	The City of Gold	ABROAD	Arabic	<null>
25	25	Rio de Janeiro	The Marvelous City	ABROAD	Portuguese	<null>
26	26	Cape Town	The Mother City	ABROAD	English	<null>
27	27	Johannesburg	The City of Gold	ABROAD	Afrikaans	<null>
28	28	São Paulo	The Capital of São Paulo	ABROAD	Portuguese	<null>
29	29	Lisbon	The Capital of Portugal	ABROAD	Portuguese	<null>
30	30	Bangkok	The Capital of Thailand	ABROAD	Thai	<null>
31	31	China	Great wall	ABROAD	Chinese	<null>

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53
4	4	Inserted a reservation.	AT000051	2023-09-16 14:29:05
5	5	Updated a reservation.	AT000051	2023-09-16 14:30:25
6	6	Deleted a reservation.	AT000051	2023-09-16 14:31:34
7	7	Inserted an event.	AT000051	2023-09-16 14:42:45
8	8	Deleted an event.	AT000051	2023-09-16 14:43:02
9	9	Inserted an event.	AT000051	2023-09-16 14:45:07
10	10	Updated an event.	AT000051	2023-09-16 14:46:59
11	11	Deleted an event.	AT000051	2023-09-16 14:48:04
12	12	Deleted a travelto.	AT000051	2023-09-16 15:05:31
13	13	Inserted a travelto.	AT000051	2023-09-16 15:08:28
14	14	Updated a travelto.	AT000051	2023-09-16 15:13:05
15	15	Inserted a destination.	AT000051	2023-09-16 15:25:14
16	16	Updated a destination.	AT000051	2023-09-16 15:26:48

DELETE

```
delete from destination where dst_id = 31;
```

	dst_id	dst_name	dst_descrc	dst_rtype	dst_language	dst_location
7	7	San Antonio	The Alamo City	LOCAL	English	<null>
8	8	San Diego	America's Finest City	LOCAL	English	<null>
9	9	Dallas	Big D	LOCAL	English	<null>
10	10	San Jose	The Capital of Silicon Valley	LOCAL	English	<null>
11	11	Athens	The Capital of Greece	LOCAL	Greek	7
12	12	Thessaloniki	The Second City of Greece	LOCAL	Greek	7
13	13	Rome	The Eternal City	ABROAD	Italian	1
14	14	Paris	The City of Love	ABROAD	French	2
15	15	London	The Capital of the UK	ABROAD	English	3
16	16	Madrid	The Capital of Spain	ABROAD	Spanish	4
17	17	Berlin	The Capital of Germany	ABROAD	German	5
18	18	Tokyo	The Capital of Japan	ABROAD	Japanese	6
19	19	Barcelona	The Capital of Catalonia	ABROAD	Spanish	<null>
20	20	Amsterdam	The Capital of the Netherlands	ABROAD	Dutch	<null>
21	21	Sydney	The Capital of Australia	ABROAD	English	<null>
22	22	Beijing	The Capital of China	ABROAD	Chinese	<null>
23	23	Moscow	The Capital of Russia	ABROAD	Russian	<null>
24	24	Dubai	The City of Gold	ABROAD	Arabic	<null>
25	25	Rio de Janeiro	The Marvelous City	ABROAD	Portuguese	<null>
26	26	Cape Town	The Mother City	ABROAD	English	<null>
27	27	Johannesburg	The City of Gold	ABROAD	Afrikaans	<null>
28	28	São Paulo	The Capital of São Paulo	ABROAD	Portuguese	<null>
29	29	Lisbon	The Capital of Portugal	ABROAD	Portuguese	<null>
30	30	Bangkok	The Capital of Thailand	ABROAD	Thai	<null>

	logid	logdescrc	log_ID_AT	log_date
1	1	Inserted a trip	AT000051	2023-09-16 14:01:52
2	2	Updated a trip.	AT000051	2023-09-16 14:04:10
3	3	Deleted a trip.	AT000051	2023-09-16 14:18:53
4	4	Inserted a reservation.	AT000051	2023-09-16 14:29:05
5	5	Updated a reservation.	AT000051	2023-09-16 14:30:25
6	6	Deleted a reservation.	AT000051	2023-09-16 14:31:34
7	7	Inserted an event.	AT000051	2023-09-16 14:42:45
8	8	Deleted an event.	AT000051	2023-09-16 14:43:02
9	9	Inserted an event.	AT000051	2023-09-16 14:45:07
10	10	Updated an event.	AT000051	2023-09-16 14:46:59
11	11	Deleted an event.	AT000051	2023-09-16 14:48:04
12	12	Deleted a travelto.	AT000051	2023-09-16 15:05:31
13	13	Inserted a travelto.	AT000051	2023-09-16 15:08:28
14	14	Updated a travelto.	AT000051	2023-09-16 15:13:05
15	15	Inserted a destination.	AT000051	2023-09-16 15:25:14
16	16	Updated a destination.	AT000051	2023-09-16 15:26:48
17	17	Deleted a destination.	AT000051	2023-09-16 15:28:23

3.3 Triggers 2

Trigger ο οποίος θα αποτρέπει την αλλαγή της ημερομηνίας αναχώρησης, ημερομηνίας επιστροφής και του κόστους του ταξιδιού, αν έχουν ήδη γίνει κρατήσεις γι' αυτό

Κώδικας

```
#second trigger
drop trigger if exists changeDateAndCost;

delimiter $

create trigger changeDateAndCost before update on trip
for each row
begin
    declare tempSeatsValue int;

    select count(*) into tempSeatsValue
    from trip
    inner join reservation r on trip.tr_id = r.res_tr_id
    where tr_id=new.tr_id
    group by res_tr_id;

    if(tempSeatsValue>0) then
        if(new.tr_cost <> old.tr_cost) then
            signal sqlstate value '45000'
            set message_text = 'You can not change the cost for this trip';
        end if;
        if(new.tr_departure <> old.tr_departure) then
            signal sqlstate value '45000'
            set message_text = 'You can not change the departure date for this trip';
        end if;
        if(new.tr_return <> old.tr_return) then
            signal sqlstate value '45000'
            set message_text = 'You can not change the return date for this trip';
        end if;
    end if;
end $
delimiter ;
```

Παράδειγμα

Στον πίνακα reservation υπάρχουν τα trips tr_id = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

```
update trip set tr_departure = '2022-01-01',tr_return = '2022-01-05', tr_cost
= 70000 where tr_id = 1 ;
```

```
travelagency> update trip set tr_departure = '2022-01-01',tr_return = '2022-01-05', tr_cost = 70000 where tr_id = 1
[2023-09-16 15:43:38] [45000][1644] You can not change the cost for this trip
[2023-09-16 15:43:38] [HY000][1644] You can not change the cost for this trip
```

3.4 Triggers 3

Trigger ο οποίος δεν θα επιτρέπει τη μείωση του μισθού ενός υπαλλήλου.






Κώδικας

```
#third trigger
drop trigger if exists changeWorkerSalary;

delimiter $

create trigger changeWorkerSalary before update on worker
for each row
begin
    if(new.wrk_salary<old.wrk_salary) then
        signal sqlstate value '45000'
        set message_text = 'There can be no reduction in the workers salary';
    end if;
end $
DELIMITER ;
```






Παράδειγμα

	 wrk_AT	÷	 wrk_name	÷	 wrk_lame	÷	 wrk_salary	÷	 wrk_br_code	÷
1	AT000001		John		Doe		45000		1	

```
update worker set wrk_salary= 10000 where wrk_AT = 'AT000001' ;
```

```
travelagency> update worker set wrk_salary= 10000 where wrk_AT = 'AT000001'
[2023-09-16 15:53:32] [45000][1644] There can be no reduction in the workers salary
[2023-09-16 15:53:32] [HY000][1644] There can be no reduction in the workers salary
```

```
update worker set wrk_salary = 60000 where wrk_AT = 'AT000001' ;
```

	 wrk_AT	÷	 wrk_name	÷	 wrk_lame	÷	 wrk_salary	÷	 wrk_br_code	÷
1	AT000001		John		Doe		60000		1	

4 Υλοποίηση GUI

Η λειτουργία του Γραφικού Περιβάλλοντος βασίζεται σε 4 κλάσεις. Τις ConnectionToDb, Login, Menu και TravelAgency. Ο σκοπός κάθε κλάσης και οι λειτουργίες που επιτελεί περιγράφονται στην συνέχεια.

4.1 TravelAgency

Η κλάση TravelAgency είναι η κλάση που περιέχει την main μέθοδο της εφαρμογής και που πηροτοδεί την εκκίνηση της εφαρμογής. Ο σκοπός αυτής της κλάσης είναι να δημιουργεί ένα αντικείμενο της κλάσης Login εμφανίζοντας στην οθόνη το αντίστοιχο παράθυρο.

4.2 ConnectionToDb

Η κλάση αυτή παρέχει την στατική μέθοδο connectMySQLDB η οποία επιστρέφει ως αντικείμενο, το αποτέλεσμα της εκτέλεσης της μεθόδου getConnection() της κλάσης DriverManager. Με την κλήση αυτής της μεθόδου πραγματοποιείται η σύνδεση με τη βάση όπου απαιτηθεί, με εύκολο τρόπο, καθώς τα στοιχεία που απαιτούνται για τη σύνδεση δίνονται μία φορά ως τιμές στις private μεταβλητές URL, USERNAME και PASSWORD της κλάσης αυτής, οι οποίες είναι επιπλέον static και final.

4.3 Login

Κατά την εκκίνηση του προγράμματος εμφανίζεται το παράθυρο Login, και ζητείται από τον χρήστη να εισάγει το όνομα χρήστη και τον κωδικό του. Η πρόσβαση επιτρέπεται μόνο σε workers που είναι itOfficers. Για τον έλεγχο των στοιχείων χρησιμοποιείται η stored procedure getItID. Η κλήση της procedure αυτής γίνεται μέσω αντικειμένου που υλοποιεί την διασύνδεση CallableStatement. Στην περίπτωση όπου τα στοιχεία είναι σωστά και η μεταβλητή εξόδου της procedure έχει τιμή διάφορη του null δημιουργείται ένα αντικείμενο της κλάσης Menu. Το αντίστοιχο παράθυρο εμφανίζεται έπειτα στην οθόνη ενώ το παράθυρο της κλάσης Login παύει να είναι ορατό. Σε αντίθετη περίπτωση, όταν η τιμή είναι null, εμφανίζεται ένα αναδυόμενο παράθυρο με το μήνυμα "Incorrect username or password!".

Κώδικας για την σύνδεση:

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try {  
        //Establish connection to the database  
        Connection con = ConnectionToDb.connectMySQLDB();  
        CallableStatement cStmt = con.prepareCall("{call getItID(?, ?, ?)}");  
        cStmt.setString(1, jTextField1.getText());  
        char[] pass = jPasswordField1.getPassword();  
        String password = new String(pass);  
        cStmt.setString(2, password);  
        cStmt.registerOutParameter(3, Types.VARCHAR);  
        cStmt.execute();  
  
        String loginResult = cStmt.getString(3);  
        //Check if itOfficer or not and proceed accordingly  
        if (loginResult != null){  
            // edw kanei switch to panel  
            Menu m = new Menu();  
            m.setTitle("IT: " + loginResult);
```



```

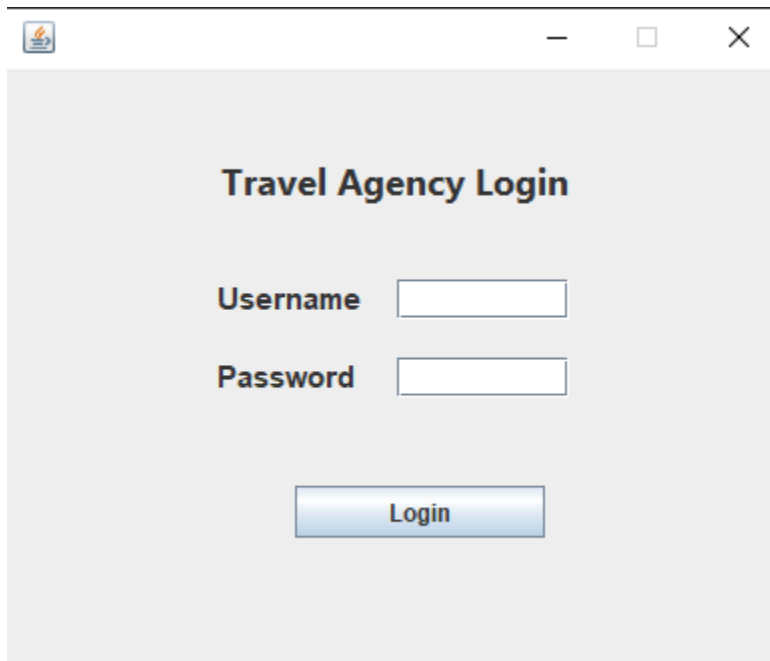
        m.setVisible(true);
        dispose();
    }else {
        //edw petaei error message kai stamataei
        JFrame f;
        f=new JFrame();
        JOptionPane.showMessageDialog(f, "Incorrect username or
password!", "Error", JOptionPane.ERROR_MESSAGE);
    }
    //Close connection
    con.close();

}catch (SQLException a){
    System.out.print(a);
}
}

```

Παράδειγμα:

- 1) Παραθύρου εισόδου



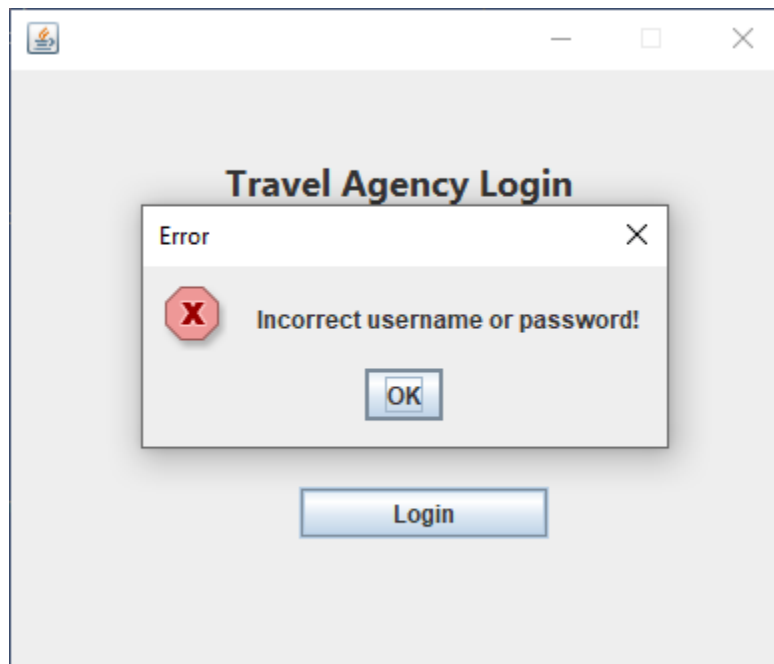
Travel Agency Login

Username

Password

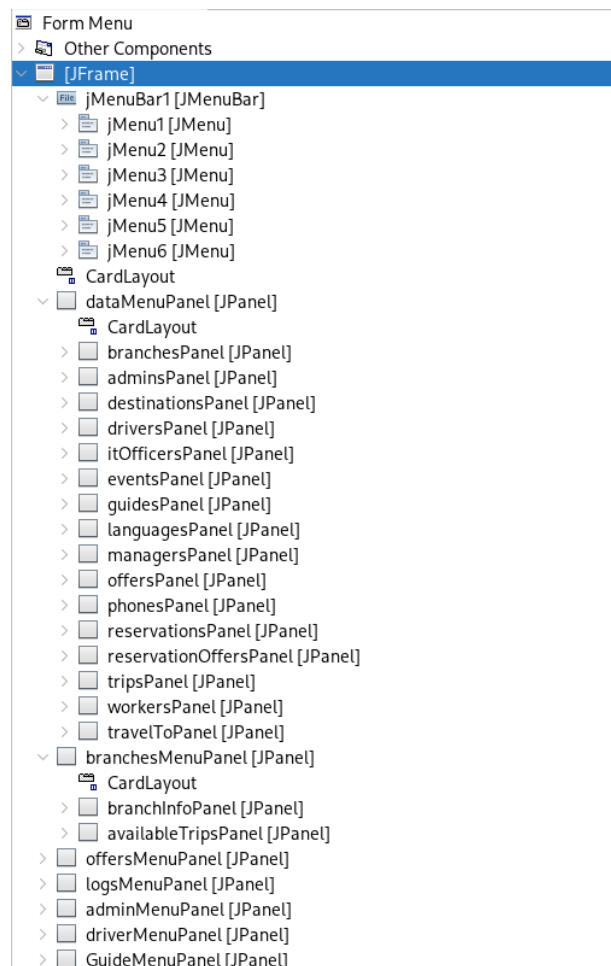
Login

2) Λάθος κωδικός χρήστη.



4.4 Menu

Η κλάση Menu είναι αυτή που παρέχει όλη την λειτουργικότητα της εφαρμογής. Το γραφικό περιβάλλον είναι δομημένο ως ένα JFrame που στην κορυφή έχει το μενού επιλογών, και από κάτω ακριβώς εμφανίζονται τα panel που αντιστοιχούν σε κάθε επιλογή. Η οργάνωση των panel γίνεται με ένα CardLayout. Κάθε λειτουργία που ζητείται να υλοποιηθεί, καθώς και οι επιπλέον λειτουργικότητες οι οποίες περιγράφονται στο επόμενο κεφάλαιο, έχουν κατηγοριοποιηθεί στο μενού αυτό. Τα πάνελ είναι οργανωμένα όπως διακρίνεται στην παρακάτω εικόνα.



Η υλοποίηση του μενού έχει γίνει με ένα αντικείμενο του τύπου jMenuBar. Κάθε κατηγορία αντιστοιχεί σε ένα jMenu και κάθε υποκατηγορία (λειτουργία) σε JMenuItem.

Data Branches Offers Logs Workers Log Out

Στη συνέχεια θα περιγράψουμε τη δομή του μενού και τις λειτουργίες που αφορούν κάθε κατηγορία του μενού, ξεκινώντας από την κατηγορία **Data**. Η κατηγορία αυτή φορά τις λειτουργίες εισαγωγής, ενημέρωσης και διαγραφής δεδομένων από όλους τους πίνακες ενδιαφέροντος. Γενικά η εμφάνιση των κατάλληλων πάνελ γίνεται με διαχείριση γεγονότων τύπου actionPerformed στα JMenuItem. Επίσης με κατάλληλο κώδικα που υπάγεται στη διαχείριση των event αυτών λαμβάνουμε σε κάποιες περιπτώσεις και δεδομένα από τη βάση, που εισάγονται για παράδειγμα σε combobox ώστε να περιορίζεται όσον το δυνατόν περισσότερο η εισαγωγή δεδομένων από τον χρήστη. Ως παράδειγμα υλοποίησης των διαδικασιών που μόλις περιεγράφηκαν παραθέτουμε τον κώδικα για τη διαχείριση event αυτού του είδους στο JMenuItem2 που αφορά το πάνελ adminsPanel.

```
private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt)
{
    //GEN-FIRST:event_jMenuItem2ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(adminsPanel).setVisible(true);

    jComboBox4.removeAllItems();
    jComboBox5.removeAllItems();

    try {
        Class.forName("com.mysql.jdbc.Driver");
        Connection con = DriverManager.getConnection(
            "jdbc:mysql://localhost:3306/travelagency","root","root");
        Connection con = ConnectionToDb.connectMySqlDB();
        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from admin");
        DefaultTableModel tbModel =
            (DefaultTableModel) jTable8.getModel();
        tbModel.setRowCount(0);
        while(rs.next()) {

            String adm_AT = rs.getString("adm_AT");
            String adm_type = rs.getString("adm_type");
            String adm_diploma = rs.getString("adm_diploma");

            String tbData[] = {adm_AT, adm_type, adm_diploma};
            //DefaultTableModel tbModel =
            (DefaultTableModel) jTable2.getModel();

            tbModel.addRow(tbData);
        }

        rs=stmt.executeQuery("select distinct wrk_AT from worker");
        while(rs.next()) {

            String br_code = rs.getString("wrk_AT");
```

```

        jComboBox4.addItem(br_code);
    }

    rs=stmt.executeQuery("select distinct adm_type from admin");
    while(rs.next()){

        String admType = rs.getString("adm_type");
        jComboBox5.addItem(admType);
    }

    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
}

```

Παρατηρούμε ότι αρχικά ρυθμίζεται κατάλληλα η ορατότητα του κάθε πάνελ και έπειτα πραγματοποιείται σύνδεση με τη βάση και εκτέλεση query ώστε να λάβουμε τα επιθυμητά δεδομένα και να αρχικοποιήσουμε τόσο τους πίνακες που εμφανίζουν τις εγγραφές της βάσης όσο και τα comboBox που περιορίζουν τις εισαγωγές από τον χρήστη.

IT: AT000051

Data Branches Offers Logs Workers Log Out

adm_AT	adm_type	adm_diploma
AT000021	ADMINISTRATIVE	Bachelor's degree in Business Administration
AT000022	ADMINISTRATIVE	Bachelor's degree in Finance
AT000023	ADMINISTRATIVE	Bachelor's degree in Public Administration
AT000024	ADMINISTRATIVE	Bachelor's degree in Supply Chain Management
AT000025	ADMINISTRATIVE	Bachelor's degree in Accounting
AT000026	ADMINISTRATIVE	Bachelor's degree in Organizational Management
AT000027	ADMINISTRATIVE	Bachelor's degree in Logistics and Transportation
AT000028	ADMINISTRATIVE	Bachelor's degree in Financial Management
AT000029	ADMINISTRATIVE	Bachelor's degree in Business Administration
AT000030	ADMINISTRATIVE	Bachelor's degree in Industrial Management
AT000031	ACCOUNTING	Bachelor's degree in Health Services Administration
AT000032	ACCOUNTING	Bachelor's degree in Criminal Justice Administration
AT000033	ACCOUNTING	Bachelor's degree in Education Administration
AT000034	ACCOUNTING	Bachelor's degree in Hospitality Administration
AT000035	ACCOUNTING	Bachelor's degree in Environmental Policy and Management
AT000036	ACCOUNTING	Bachelor's degree in International Business
AT000037	ACCOUNTING	Bachelor's degree in Public Policy
AT000038	ACCOUNTING	Bachelor's degree in Project Management
AT000039	ACCOUNTING	Bachelor's degree in Social Work Administration
AT000040	ACCOUNTING	Bachelor's degree in Nonprofit Administration
AT000041	LOGISTICS	Bachelor's degree in Communication Studies
AT000042	LOGISTICS	Bachelor's degree in Information Technology Management
AT000043	LOGISTICS	Bachelor's degree in Marketing Administration
AT000044	LOGISTICS	Bachelor's degree in Retail Management
AT000045	LOGISTICS	Bachelor's degree in Sports Administration

New Admin

adm_AT

AT000001

Change Admin Data

adm_type

ADMINISTRATIVE

Delete Admin

adm_diploma

Όπως διακρίνεται από την παραπάνω εικόνα, για κάθε λειτουργία υπάρχει και ένα αντίστοιχο κουμπί, για τα οποία υπάρχουν πάλι listeners για event τύπου actionPerformed. Για την εισαγωγή μίας εγγραφής αρκεί η σωστή συμπλήρωση όλων των πεδίων. Για την τροποποίηση μίας εγγραφής χρειάζεται να επιλεγεί η αντίστοιχη εγγραφή από τον πίνακα των εγγραφών και να εισαχθούν τα επιθυμητά δεδομένα στα πεδία (τουλάχιστον σε ένα πεδίο). Η διαγραφή επιτυγχάνεται με την επιλογή της εγγραφής από τον πίνακα και “πατήματος” του κουμπιού διαγραφής. Ένα απλό παράδειγμα είναι η εισαγωγή ενός admin:

```
private void jButton18ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton18ActionPerformed
    // TODO add your handling code here:

    if( !jTextField21.getText().equals("") ){

        String adm_AT = jComboBox4.getSelectedItem().toString();
        String adm_type = jComboBox5.getSelectedItem().toString();
        String adm_diploma = jTextField21.getText();

        String insertAdminCommand = "INSERT INTO admin VALUES ('" + adm_AT +
        "','" + adm_type + "','" + adm_diploma + "');"

        try {
            Connection con = ConnectionToDb.connectMySqlDB();

            Statement stmt=con.createStatement();
            stmt.executeUpdate(insertAdminCommand);

            DefaultTableModel tbModel =
            (DefaultTableModel)jTable8.getModel();
            String tbData[] = { adm_AT , adm_type , adm_diploma };
            tbModel.addRow(tbData);
            con.close();
        }
        catch(Exception e){ JFrame f=new JFrame();
        JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

    }
}
```

Η διαγραφή ενός admin είναι ένα αντιπροσωπευτικό παράδειγμα των περιπτώσεων που υπάρχουν περιορισμοί στις ενέργειες τροποποίησης πινάκων της βάσης που μπορεί να εκτελέσει ένας itOfficer και οι οποίες απαιτούν την χρήση procedure. (σύμφωνα με τις απαιτήσεις του project). Παρόμοια περίπτωση είναι η εισαγωγή itOfficer.

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton3ActionPerformed
    // TODO add your handling code here:
```

```

String at = jTextField34.getText();
String fn = jTextField45.getText();
String ln = jTextField46.getText();
String sal = jTextField47.getText();
String bCode = jComboBox29.getSelectedItem().toString();
String pass = jTextField49.getText();
String sd = jTextField50.getText();
String ed = jTextField51.getText();
String proc = "call
addIt('"+at+"', '"+fn+"', '"+ln+"', '"+sal+"', '"+bCode+"', '"+pass+"', '"+sd+"', '"+ed+"');";
System.out.println(proc);
try{
    Connection con = ConnectionToDb.connectMySQLDB();
    CallableStatement stmt = con.prepareCall(proc);
    ResultSet rs = stmt.executeQuery();

    DefaultTableModel itOfficerModel =
(DefaultTableModel) jTable15.getModel();
    String tbData[] = {at, fn, ln, sal, bCode, pass, sd, ed};
    itOfficerModel.addRow(tbData);

    con.close();
}catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
}

```

Η μείωση μισθού εργαζομένου είναι ένα αντιπροσωπευτικό παράδειγμα των περιπτώσεων που υπάρχουν περιορισμοί στις ενέργειες τροποποίησης πινάκων της βάσης που μπορεί να εκτελέσει ένας itOfficer και οι οποίες “πυροδοτούν” κάποιον trigger. Για την υλοποίησή τους, αρκεί να “πιάσουμε” το SQLException και να τυπώσουμε το αντίστοιχο μήνυμα που έχει οριστεί από τον trigger για το συγκεκριμένο sqlState value σε ένα JOptionPane. Παρόμοια είναι η περίπτωση τροποποίησης στοιχείων ταξιδιού αν υπάρχουν κρατήσεις.

```

private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-
FIRST:event_jButton7ActionPerformed
// TODO add your handling code here:

int selectedRow = jTable4.getSelectedRow();
if(selectedRow != -1){

    String key1 = jTable4.getValueAt(selectedRow, 0).toString();
    String key2 = jTable4.getValueAt(selectedRow, 4).toString();

```

```

try {

    Connection con = ConnectionToDb.connectMySQLDB();

    String[] data = new String[ 5 ];

    data[0] = jTextField6.getText();
    data[1] = jTextField7.getText();
    data[2] = jTextField8.getText();
    data[3] = jTextField9.getText();
    data[4] = jComboBox2.getSelectedItem().toString();

    String command1 = "UPDATE worker set wrk_name = '" + data[1]
+ "' where wrk_AT = '" + key1 + "' and wrk_br_code = '" + key2 + "' ";
    String command2 = "UPDATE worker set wrk_lame = '" + data[2]
+ "' where wrk_AT = '" + key1 + "' and wrk_br_code = '" + key2 + "' ";
    String command3 = "UPDATE worker set wrk_salary = '" + data[3]
+ "' where wrk_AT = '" + key1 + "' and wrk_br_code = '" + key2 + "' ";
    String command4 = "UPDATE worker set wrk_br_code = '" +
data[4] + "' where wrk_AT = '" + key1 + "' and wrk_br_code = '" + key2 + "' ";

    if(!data[1].equals("")){
        CallableStatement stmt = con.prepareCall(command1);
        stmt.executeUpdate();
    }
    if(!data[2].equals("")){
        CallableStatement stmt = con.prepareCall(command2);
        stmt.executeUpdate();
    }
    if(!data[3].equals("")){
        CallableStatement stmt = con.prepareCall(command3);
        stmt.executeUpdate();
    }
    if(!data[4].equals("")){
        CallableStatement stmt = con.prepareCall(command4);
        stmt.executeUpdate();
        key2 = data[4];
    }

    if(!data[0].equals("")){
        String command0 = "UPDATE worker set wrk_AT = '" +
data[0] + "' where wrk_AT = '" + key1 + "' and wrk_br_code = '" + key2 + "' ";
        CallableStatement stmt = con.prepareCall(command0);
        stmt.executeUpdate();
    }

    //Update table
    CallableStatement stmt = con.prepareCall("select * from
worker");

```



```

        ResultSet rs= stmt.executeQuery();
        DefaultTableModel tbModel =
(DefaultTableModel) jTable4.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String wrk_AT = rs.getString("wrk_AT") ;
            String wrk_name = rs.getString("wrk_name") ;
            String wrk_lame = rs.getString("wrk_lame") ;
            String wrk_salary = rs.getString("wrk_salary") ;
            String wrk_br_code = rs.getString("wrk_br_code") ;

            String tbData[] = {wrk_AT, wrk_name ,wrk_lame ,wrk_salary
, wrk_br_code};

            tbModel.addRow(tbData);

        }

        con.close();
    }
    catch(SQLException e){
        JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e.getMessage(),"Alert",JOptionPane.WARNING_ME
SSAGE);
    }
}
}

```

Οι λειτουργίες που αφορούν τα υποκαταστήματα και βασίζονται σε procedures ανήκουν στην κατηγορία Branches, όπου έχουμε δύο επιλογές. Η πρώτη αφορά τις procedures showBranchDetails και branchInfo, τα αποτελέσματα των οποίων εμφανίζονται στο ίδιο panel. Η δεύτερη κατηγορία αφορά την procedure showAvailableTrips, η οποία εμφανίζεται στο δικό της ξεχωριστό panel. Η procedure του ερωτήματος 3.1.3.2 ανήκει στην κατηγορία Offers. Ο πίνακας log στον οποίο καταγράφονται όλες οι ενέργειες του ItOfficer ανήκει στην κατηγορία Offers. Οι διαδικασίες υλοποίησης των προαναφερθέντων είναι όμοιες με όσες έχουν ήδη περιγραφεί.

Τέλος στο μενού έχει προστεθεί και η επιλογή για log out. Καταγράφονται event mousePressed για το μενού αυτό ώστε να εμφανιστεί κατάλληλο JOptionPane. Στην περίπτωση που ο χρήστης επιλέξει yes, τότε εγκαθίσταται σύνδεση με τη βάση, διαγράφονται οι εγγραφές του πίνακα tempLogin, ώστε να μην παραμείνουν τα στοιχεία του χρήστη στον πίνακα αυτό μετά την έξοδό του, το πάνελ του Menu γίνεται μη ορατό και εμφανίζεται το παράθυρο της σύνδεσης του χρήστη.

```

private void jMenuItem6MousePressed(java.awt.event.MouseEvent evt) { //GEN-
FIRST:event_jMenuItem6MousePressed
    // TODO add your handling code here:
    JFrame logoutFrame = new JFrame();
    int choice = JOptionPane.showConfirmDialog(logoutFrame, "Are you sure you

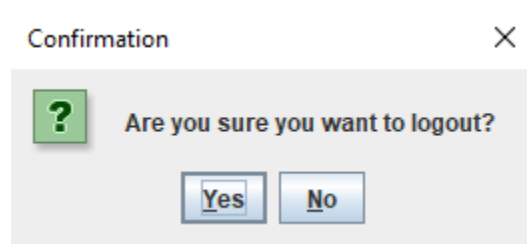
```

```

want to logout?", "Confirmation", JOptionPane.YES_NO_OPTION);
if (choice == JOptionPane.YES_OPTION) {
    try{
        Connection con = ConnectionToDb.connectMySQLDB();

        //Delete temporary
        Statement stmt3=con.createStatement();
        stmt3.execute("truncate table templogin;");
        dispose();
        new Login().setVisible(true);
    }catch(Exception e){System.out.println(e);}
}else
    logoutFrame.dispose();
}

```



Επιπλέον για να σιγουρευτούμε πως κατά την έξοδο από το πρόγραμμα δεν θα παραμείνει εγγραφή στον πίνακα tempLogin υπάρχει listener για event τύπου closingWindow για το κύριο frame της κλάσης Menu ώστε και πάλι να διαγράφεται η όποια εγγραφή.

```

private void formWindowClosing(java.awt.event.WindowEvent evt) { //GEN-FIRST:event_formWindowClosing
    // TODO add your handling code here:
    JFrame exitFrame = new JFrame();
    int choice = JOptionPane.showConfirmDialog(exitFrame,
        "Are you sure you want to close this window?", "Close Window?",
        JOptionPane.YES_NO_OPTION,
        JOptionPane.QUESTION_MESSAGE);
    if (choice == JOptionPane.YES_OPTION) {
        try{
            Connection con = ConnectionToDb.connectMySQLDB();

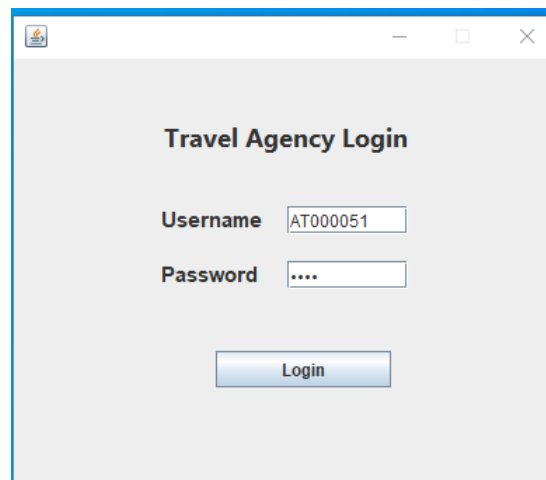
            //Delete temporary
            Statement stmt=con.createStatement();
            stmt.execute("truncate table templogin;");
            System.exit(0);

        }catch(Exception e){System.out.println(e);}
        System.exit(0);
    }
}

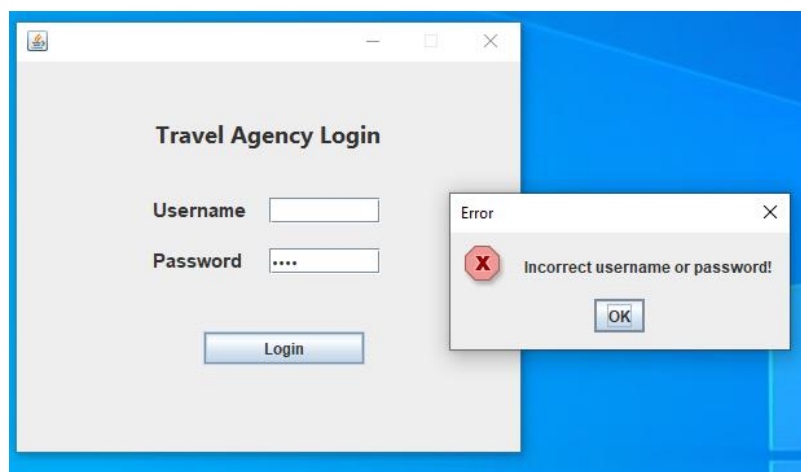
```

```
}else  
    exitFrame.dispose();  
}
```

4.5 Παραδείγματα



Login Exception



Επιλογή πίνακα μέσω του μενού

IT: AT000051

Data

Branches

Offers

Logs

Workers

Log Out

Admins

Branches

Destinations

Drivers

Events

Guides

It Officers

Languages

Managers

Offers

Phones

Reservations

Reservation offers

Trips

Workers

TravelTo

br_code	br_num	br_street	br_city
1		1 Main Street	New York
2		2 Maple Avenue	Chicago
3		3 1st Street	Los Angeles
4		4 2nd Avenue	Houston
5		5 3rd Boulevard	Philadelphia
6		6 14th Street	Phoenix
7		7 25th Avenue	San Antonio
8		8 36th Boulevard	San Diego
9		9 47th Street	Dallas
10		10 58th Avenue	San Jose
11		1 Aristotelous	Thessaloniki

New Branch

Change Branch Data

Delete Branch

br_code

11

br_num

1

br_street

Aristotelous

br_city

Thessaloniki

Εισαγωγή νέου branch

IT: AT000051

Data

Branches

Offers

Logs

Workers

Log Out

br_code	br_num	br_street	br_city
1		1 Main Street	New York
2		2 Maple Avenue	Chicago
3		3 1st Street	Los Angeles
4		4 2nd Avenue	Houston
5		5 3rd Boulevard	Philadelphia
6		6 14th Street	Phoenix
7		7 25th Avenue	San Antonio
8		8 36th Boulevard	San Diego
9		9 47th Street	Dallas
10		10 58th Avenue	San Jose
11		1 Aristotelous	Thessaloniki

New Branch

Change Branch Data

Delete Branch

br_code

11

br_num

1

br_street

Aristotelous

br_city

Thessaloniki

Update Branch

IT: AT000051

Data

Branches

Offers

Logs

Workers

Log Out

br_code	br_num	br_street	br_city
1	1	Main Street	New York
2	2	Maple Avenue	Chicago
3	3	1st Street	Los Angeles
4	4	2nd Avenue	Houston
5	5	3rd Boulevard	Philadelphia
6	6	14th Street	Phoenix
7	7	25th Avenue	San Antonio
8	8	36th Boulevard	San Diego
9	9	47th Street	Dallas
10	10	58th Avenue	San Jose
11	1	Tinou	Patra

New Branch

Change Branch Data

Delete Branch

br_code

11

br_num

1

br_street

Tinou

br_city

Patra

Delete Branch μετά από επιλογή.

IT: AT000051

Data

Branches

Offers

Logs

Workers

Log Out

br_code	br_num	br_street	br_city
1	1	Main Street	New York
2	2	Maple Avenue	Chicago
3	3	1st Street	Los Angeles
4	4	2nd Avenue	Houston
5	5	3rd Boulevard	Philadelphia
6	6	14th Street	Phoenix
7	7	25th Avenue	San Antonio
8	8	36th Boulevard	San Diego
9	9	47th Street	Dallas
10	10	58th Avenue	San Jose

New Branch

Change Branch Data

Delete Branch

br_code

br_num

br_street

br_city

Προσπάθεια αλλαγής δεδομένων του trip που είναι σε reservation.

IT: AT000051

Data Branches Offers Logs Workers Log Out

tr_id	tr_departure	tr_return	tr_maxseats	tr_cost	tr_br_code	tr_gul_AT	tr_drv_AT
1	2022-01-01	2022-01-07	30	1000.0	1	AT000001	AT000011
2	2022-01-08	2022-01-14	35	1200.0	2	AT000002	AT000012
3	2022-01-15	2022-01-21	40	1500.0	3	AT000003	AT000013
4	2022-01-22	2022-01-29	45	1700.0	4	AT000004	AT000014
5	2022-01-29	2022-02-04	50	2000.0	5	AT000005	AT000015
6	2022-02-05	2022-02-11	55	2200.0	6	AT000006	AT000016
7	2022-02-12	2022-02-19	60	2500.0	7	AT000007	AT000017
8	2022-02-19	2022-02-25	65	2700.0	8	AT000008	AT000018
9	2022-02-26	2022-03-04	70	3000.0	9	AT000009	AT000019
10	2022-03-05	2022-03-11	75	3200.0	10	AT000010	AT000020
11	2022-03-12	2022-03-19	80	3500.0	1	AT000001	AT000011
12	2022-03-19	2022-03-25	85	3700.0	2	AT000002	AT000012
13	2022-03-26	2022-04-01	90	4000.0	3	AT000003	AT000013
14	2022-04-02	2022-04-08	95	4200.0	4	AT000004	AT000014
15	2022-04-09	2022-04-15	100	4500.0	5	AT000005	AT000015
16	2022-04-16	2022-04-22	105	4700.0	6	AT000006	AT000016
17	2022-04-23	2022-04-29	110	5000.0	7	AT000007	AT000017
18	2022-04-30	2022-05-06	115	5200.0	8	AT000008	AT000018
19	2022-05-07	2022-05-13	120	1200.0	9	AT000009	AT000019
20	2022-05-14	2022-05-20	125	1100.0	10	AT000010	AT000020
21	2022-05-21	2022-05-27	30	1000.0	1	AT000001	AT000011
22	2022-05-28	2022-06-03	35	900.0	2	AT000002	AT000012
23	2022-06-04	2022-06-10	40	800.0	3	AT000003	AT000013
24	2022-06-11	2022-06-17	45	700.0	4	AT000004	AT000014
25	2022-06-18	2022-06-24	50	600.0	5	AT000005	AT000015

New trip tr_id tr_cost 5000.0

Change trip data tr_departure tr_br_code 1

Delete trip tr_return tr_gul_AT AT000001

tr_maxseats tr_drv_AT AT000011

Alert You can not change the cost for this trip OK

Προσπάθεια μείωσης μισθών εργαζομένου

IT: AT000051

Data Branches Offers Logs Workers Log Out

wrk_AT	wrk_name	wrk_lname	wrk_salary	wrk_br-code
AT000001	John	Doe	45000.0	1
AT000002	Jane	Doe	50000.0	2
AT000003	Bob	Smith	40000.0	3
AT000004	Sue	Smith	55000.0	4
AT000005	Alice	Jones	52000.0	5
AT000006	Mike	Jones	48000.0	6
AT000007	Sarah	Williams	45000.0	7
AT000008	Chris	Williams	52000.0	8
AT000009	Dave	Johnson	50000.0	9
AT000010	Emily			10
AT000011	Laura			1
AT000012	Tom			2
AT000013	Kim			3
AT000014	Steve			4
AT000015	Julie			5
AT000016	Jason			6
AT000017	Samantha			7
AT000018	Mark	Miller	48000.0	8
AT000019	Alex	Wilson	45000.0	9
AT000020	Megan	Wilson	52000.0	10
AT000021	Tina	Taylor	50000.0	1
AT000022	Ryan	Taylor	45000.0	2
AT000023	Rachel	Anderson	52000.0	3
AT000024	Adam	Anderson	48000.0	4
AT000025	Katie	Thomas	45000.0	5
AT000026	Scott	Thomas	52000.0	6
AT000027	Ben	Johnson	50000.0	7

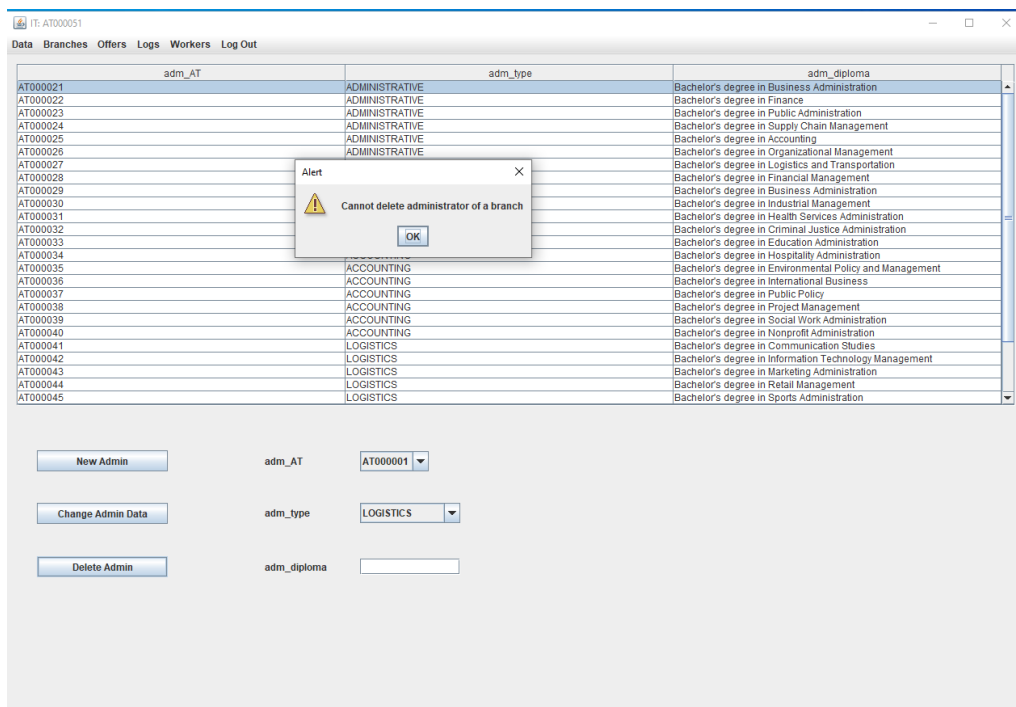
New Worker wrk_AT

Change Worker Data wrk_name wrk_lname wrk_salary 30000.0 wrk_br-code 1

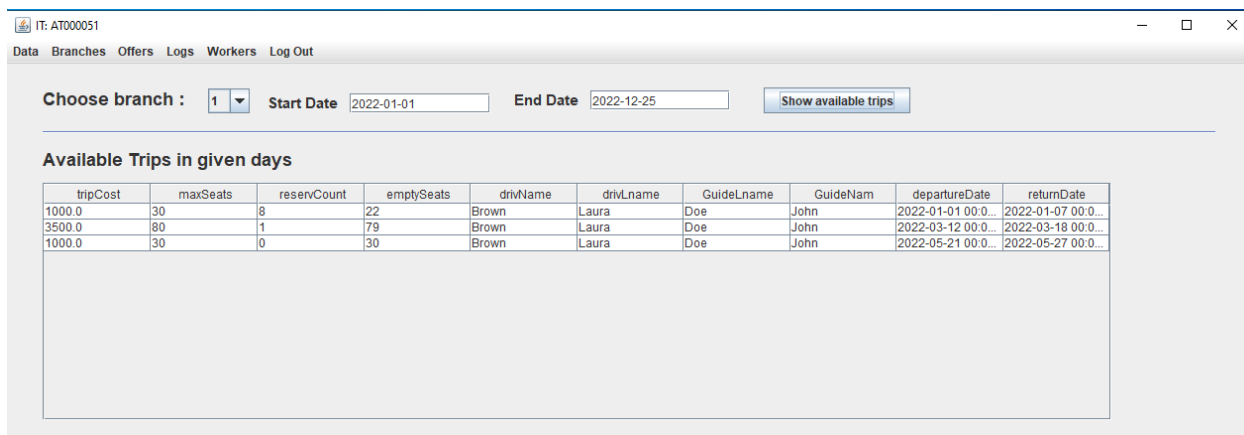
Delete Worker

Alert There can be no reduction in the workers salary OK

Προσπάθεια διαγραφής administrator σε branch




Στο παράθυρο Available Trips για κάθε υποκατάστημα και για ένα διάστημα που ορίζει ο ίδιος φαίνεται, το κόστος ταξιδιού, μέγιστες θέσεις ανά ταξίδι, σύνολο κρατήσεων (reservation), διαθέσιμες κενές θέσεις, επώνυμο και όνομα οδηγού και ξεναγού, ημερομηνία αναχώρησης και επιστροφής



Στο Branch Info παράθυρο βλέπουμε στο Branch Info πίνκα για κάθε υποκατάστημα τα στοιχεία του, το ονοματεπώνυμο του διευθυντή του, το σύνολο κρατήσεων και το σύνολο εσόδων (κράτηση επί κόστος ταξιδιού).

Στον πίνακα Financial τα έσοδα, τα έξοδα και τον τζίρο του branch.

Στον πίνακα Worker's Info βλέπουμε τα δεδομένα των εργαζομένων του branch και τον συνολικό μισθό εργαζομένων.

 IT: AT000051

DataBranchesOffersLogsWorkersLog Out

Choose branch : 1

Branch Info

Branch Code	Number	Street	City	admin (FN)	Last name (LN)	Num of reservations	Income
1	1	Main Street	New York	Tina	Taylor	9	11500.0

Financial

Income	Expenses	Turnover
11500	322000	-310500

Worker's info

First Name	Last Name	Salary
John	Doe	45000.0
Laura	Brown	52000.0
Tina	Taylor	50000.0
George	Bakris	12000.0
Thanasis	Bakalis	12000.0
Haris	Stamelos	12000.0
Maria	Papadopoulou	46000.0
Marios	Antoniou	48000.0
Stavros	Nikolaidis	45000.0

Total salary of workers
322000.0

Στο παράθυρο Offers Procedure βλέπουμε ότι δίνουμε το επώνυμο ενός πελάτη και εμφανίζονται τα ονόματα και επώνυμα των πελατών με το επώνυμο αυτό και την προσφορά ταξιδιού στην οποία έχει γίνει εγγραφή. Σε περίπτωση που υπάρχουν πάνω από ένας εμφανίζεται και το πλήθος των πελατών με το όνομα αυτό, ανά προσφορά ταξιδιού.

IT: AT000051

Data

Branches

Offers

Logs

Workers

Log Out

Customer Name:

Show reservations

res_name	res_lname	offer_id	offer_startDate	offer_endDate	cost_per_person	offer_dst_id

offer_id	offer_startDate	offer_endDate	cost_per_person	offer_dst_id	count(res_lname)

Πίνακας Logs

IT: AT000051

Data

Branches

Offers

Logs

Workers

Log Out

LOGS

Description	IT's Last Name	log_date
Inserted an event.	Papaioannou	2023-09-16 21:14:44
Deleted an event.	Papaioannou	2023-09-16 21:15:39

4.6 Procedures για έξτρα λειτουργίες

```
drop procedure if exists showBranchDetails;

delimiter $
CREATE PROCEDURE showBranchDetails(brCode INT)
BEGIN
    declare not_found int;
    declare temp_tr_id int;
    declare temp_cost float(7, 2);
    declare total_cost float(7, 2);

    -- total income from the reservations
    declare trip_cost_cursor cursor for
    select tr_cost, tr_id
    from trip
    where tr_br_code = brCode;

    declare continue handler for not found set not_found=1;

    set not_found=0;
    set total_cost=0.0;
    open trip_cost_cursor;

    repeat
        fetch trip_cost_cursor INTO temp_cost, temp_tr_id;
        if(not_found=0) then
            select count(res_tr_id) * temp_cost
            from reservation
            where res_tr_id = temp_tr_id INTO temp_cost;

            set total_cost = total_cost + temp_cost;
        end if ;
        until (not_found=1)
    end repeat ;
    close trip_cost_cursor;

    -- total number of reservations
    select COUNT(ALL res_tr_id) INTO @numOfRes
    from reservation
    inner join trip t on reservation.res_tr_id = t.tr_id
    where tr_br_code = brCode;

    SELECT 'Branch Details' AS 'Result Type', br_code AS 'Code', br_num AS 'Number', br_street AS
'Street', br_city AS
'City', wrk_name AS 'admin (FN)', wrk_lame AS 'Last name (LN)', @numOfRes AS 'Num of
reservations', total_cost AS
'Income'
    FROM branch
    inner join worker w on branch.br_code = w.wrk_br_code
    inner join admin a on w.wrk_AT = a.adm_AT
    WHERE brCode = br_code and adm_type = 'ADMINISTRATIVE';
END$

delimiter ;

drop procedure if exists branchInfo;

delimiter $
CREATE PROCEDURE branchInfo (brCode INT)
BEGIN
    SELECT wrk name AS 'First Name', wrk_lame AS 'Last Name', wrk_salary AS 'Salary'
    FROM worker
    WHERE wrk_br_code = brCode;

    SELECT 'Total Salaries' AS 'Result Type', SUM(wrk_salary)
    FROM worker
```

```

        WHERE wrk_br_code = brCode;
END$

delimiter ;

drop procedure if exists addIT;
delimiter $
CREATE PROCEDURE addIT(in wrkAT char(10), in wrkName varchar(20), in wrkLname varchar(20), in
wrkSalary float(7, 2),in
        wrkBrCode int, in password char(10), in startD datetime, in endD datetime)
BEGIN
    -- First insert a worker
    insert into worker values(wrkAT, wrkName, wrkLname, wrkSalary, wrkBrCode);

    -- After insert on itofficer table
    insert into itOfficer values (wrkAT, password, startD, endD);

END $
delimiter ;

```

```

drop procedure if exists showLog;

delimiter $
CREATE PROCEDURE showLog()
BEGIN
    DECLARE not_found INT;
    DECLARE logD TEXT;
    DECLARE logID CHAR(10);
    DECLARE itLname VARCHAR(20);
    DECLARE logDate DATETIME;

    DECLARE logcursor CURSOR FOR
    SELECT logdescrc, log_ID_AT, log_date
    FROM log;

    DECLARE CONTINUE HANDLER FOR NOT FOUND SET not_found = 1;

    SET not_found = 0;
    OPEN logcursor;

    -- Create a temporary table to store the data
    CREATE TEMPORARY TABLE temp_log_result (
        log_desc TEXT,
        it_lastname VARCHAR(20),
        log_date DATETIME
    );

    FETCH_LOOP: LOOP
        FETCH logcursor INTO logD, logID, logDate;
        IF not_found = 1 THEN
            LEAVE FETCH_LOOP;
        END IF;

        SELECT wrk_lame INTO itLname
        FROM worker
        INNER JOIN itOfficer ON worker.wrk_AT = itOfficer.it_AT
        WHERE wrk_AT = logID;

        INSERT INTO temp_log_result (log_desc, it_lastname, log_date)
        VALUES (logD, itLname, logDate);
    END LOOP FETCH_LOOP;

    CLOSE logcursor;

    -- Fetch the complete result set from the temporary table
    SELECT log_desc, it_lastname, log_date FROM temp_log_result;

    -- Drop the temporary table
    DROP TEMPORARY TABLE temp_log_result;

```

```
END;
```

```
delimiter ;
```

5 Μέθοδοι επιπλέον λειτουργικότητας

Επιπλέον των βασικών λειτουργιών που περιγράφουν οι απαιτήσεις του project υλοποιήθηκαν οι ακόλουθες λειτουργίες:

- 1) Εμφάνιση για κάθε branch των συνολικών εσόδων του (κόστη ταξιδιών), εξόδων (μισθοί) και του τζίρου (κόστη ταξιδιών - μισθοί).

Ο πίνακας που επιστρέφει αυτή η διαδικασία εμφανίζεται στο Branches -> Branch Info

- 2) Εμφάνιση καρτέλας πληροφοριών για workers τύπου admin, driver και guide

Οι καρτέλες βρίσκονται στη νέα κατηγορία “Workers” του μενού και πιο συγκεκριμένα

Workers -> Admin Card, Workers -> Guide Card, Workers -> Driver Card.

Οι υλοποίηση των λειτουργιών αυτών ακολουθεί το μοτίβο των προηγούμενων λειτουργιών.

Παραδείγματα

5.1 Επιπλέον μέθοδος λειτουργικότητας 1

IT: AT000051

Data Branches Offers Logs Workers Log Out

Choose branch : 1

Branch Info

Branch Code	Number	Street	City	admin (FN)	Last name (LN)	Num of reservations	Income
1	1	Main Street	New York	Tina	Taylor	9	11500.0

Financial

Income	Expenses	Turnover
11500	322000	-310500

Worker's info

First Name	Last Name	Salary
John	Doe	45000.0
Laura	Brown	52000.0
Tina	Taylor	50000.0
George	Baknis	12000.0
Thanasis	Bakalis	12000.0
Haris	Stamelos	12000.0
Maria	Papadopoulou	46000.0
Marios	Antoniou	48000.0
Stavros	Nikolaidis	45000.0

Total salary of workers
322000.0

5.2 Επιπλέον μέθοδος λειτουργικότητας 2

Στα παράθυρα των workers μπορούμε να επιλέξουμε και να δούμε την καρτέλα κάθε εργαζομένου ανάλογα με το αν είναι admin, driver ή guide.

The screenshot shows the 'Admin' worker profile page. The title bar indicates the application is 'IT: AT000051'. The menu bar includes 'Data', 'Branches', 'Offers', 'Logs', 'Workers', and 'Log Out'. The 'Workers' menu is open, showing 'Admin', 'Driver', and 'Guide' options. The main content area is titled 'Admin' and displays the following details:

- First Name:** Tina
- Last Name:** Taylor
- AT:** AT000021
- Branch Code:** 1
- Salary:** 50000.0
- Type:** ADMINISTRATIVE

On the right, there is a 'Diploma:' section with a text box containing 'Bachelor's degree in Business Administration'.

The screenshot shows the 'Driver' worker profile page. The title bar indicates the application is 'IT: AT000051'. The menu bar includes 'Data', 'Branches', 'Offers', 'Logs', 'Workers', and 'Log Out'. The 'Workers' menu is open, showing 'Admin', 'Driver', and 'Guide' options. The main content area is titled 'Driver' and displays the following details:

- First Name:** Laura
- Last Name:** Brown
- AT:** AT000011
- Branch Code:** 1
- Salary:** 52000.0
- License:** B
- Route:** LOCAL
- Driver Exp:** 5

On the right, there is a 'Trips' section with a table showing trip data:

Trip ID	Start Date	End Date
1	2022-01-01 00:00:00	2022-01-07 00:00:00
11	2022-03-12 00:00:00	2022-03-18 00:00:00
21	2022-05-21 00:00:00	2022-05-27 00:00:00

IT: AT000051

Data
Branches
Offers
Logs
Workers
Log Out

Guide

AT000001

First Na...

John

Last Name:

Doe

AT:

AT000001

Branch Code:

1

Salary:

45000.0

Languages

French

Spanish

Trips

Trip ID	Start Date	End Date
1	2022-01-01 00:00:00	2022-01-07 00:00:00
11	2022-03-12 00:00:00	2022-03-18 00:00:00
21	2022-05-21 00:00:00	2022-05-27 00:00:00

CV

I have been working as a tour guide for 10 years and have extensive knowledge of local history and landmarks.

5.3 Επιπλέον procedures

```
-- Extra functionalities
drop procedure if exists branch_stats;

delimiter $
CREATE PROCEDURE branch_stats(brCode INT)
BEGIN
    declare not_found int;
    declare temp_tr_id int;
    declare temp_cost float(7, 2);
    declare total_cost float(7, 2);
    declare total_salaries float(13, 2);

    -- total income from the reservations
    declare trip_cost_cursor cursor for
    select tr_cost, tr_id
    from trip
    where tr_br_code = brCode;

    declare continue handler for not found set not_found=1;

    set not_found=0;
    set total_cost=0.0;
    open trip_cost_cursor;

    repeat
        fetch trip_cost_cursor INTO temp_cost, temp_tr_id;
        if(not_found=0) then
            select count(res_tr_id) * temp_cost
            from reservation
            where res_tr_id = temp_tr_id INTO temp_cost;

            set total_cost = total_cost + temp_cost;
        end if ;
    until (not_found=1)
    end repeat ;
```



```

        close trip_cost_cursor;

        -- expenses = salaries
        SELECT SUM(wrk_salary)
        FROM worker
        WHERE wrk_br_code = brCode INTO total_salaries;

        select total_cost AS 'Income', total_salaries AS 'expenses', total_cost - total_salaries AS
        'Turnover';
END $

drop procedure if exists driver_card;

delimiter $
CREATE PROCEDURE driver_card(id char(10))
BEGIN
    -- id
    select wrk_name AS "First Name", wrk_lame AS "Last Name", wrk_AT AS "AT", wrk_br_code AS
    'Branch Code', wrk_salary AS
    "Salary", drv_license AS "License", drv_route AS "Route", drv_experience AS "Experience"
    from worker
    inner join driver on drv_AT = worker.wrk_AT
    where wrk_AT = id;

    -- assigned trips
    select tr_id AS "Trip Id", tr_departure AS "departure", tr_return AS "Return"
    from trip
    where tr_drv_AT = id;
END $

drop procedure if exists guide_card;

delimiter $
CREATE PROCEDURE guide_card(id char(10))
BEGIN
    -- id
    select wrk_name AS "First Name", wrk_lame AS "Last Name", wrk_AT AS "AT", wrk_br_code AS
    'Branch Code', wrk_salary AS
    "Salary"
    from worker
    inner join guide on guide.gui_AT = worker.wrk_AT
    where wrk_AT = id;

    -- languages
    select lng_language
    from languages
    where lng_gui_AT = id;

    -- assigned trips
    select tr_id AS "Trip Id", tr_departure AS "departure", tr_return AS "Return"
    from trip
    where tr_gui_AT = id;

    -- guide cv
    select gui_cv AS "CV"
    from guide
    where gui_AT = id;
END $

drop procedure if exists admin_card;
delimiter $
CREATE PROCEDURE admin_card(id char(10))
BEGIN
    -- id
    select wrk_name AS "First Name", wrk_lame AS "Last Name", wrk_AT AS "AT", wrk_br_code AS

```

```
'Branch Code', wrk_salary AS  
    "Salary", adm_type AS "Type", adm_diploma AS 'Diploma'  
from worker  
inner join admin on adm_AT = wrk_AT  
where wrk_AT = id;  
  
END $
```

6 Κώδικας MySQL

```
drop database if exists travelagency;
create database if not exists travelagency;
use travelagency;

create table if not exists branch(
    br_code int(11) not null auto_increment ,
    br_num int(4) not null,
    br_street varchar(30) default 'unknown' not null,
    br_city varchar(30) default 'unknown' not null,
    primary key(br_code)
);

create table if not exists phones(
    ph_br_code int(11) not null,
    ph_number char(10) default 'unknown' not null,
    primary key (ph_br_code,ph_number),
    constraint branchPhone foreign key (ph_br_code) references
branch(br_code)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists destination(
    dst_id int(11) auto_increment,
    dst_name varchar(50) default 'unknown' not null,
    dst_descr text not null,
    dst_rtype enum('LOCAL','ABROAD'),
    dst_language varchar(30) default 'unknown' not null,
    dst_location int(11),
    primary key(dst_id),
    constraint locationId foreign key (dst_location) references
destination(dst_id)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists worker(
    wrk_AT char(10) default 'unknown' not null,
    wrk_name varchar(20) default 'unknown' not null,
    wrk_lame varchar(20) default 'unknown' not null,
    wrk_salary float(7,2) not null,
    wrk_br_code int(11) not null ,
    PRIMARY KEY (wrk_AT, wrk_br_code), #Ένας εργαζόμενος εργάζεται σε ένα μόνο
υποκατάστημα
    constraint workerBranch foreign key (wrk_br_code) references
branch(br_code)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists guide(
    gui_AT char(10) not null,
    gui_cv text not null,
    PRIMARY KEY (gui_AT),
    constraint workerGuide foreign key (gui_AT) references worker(wrk_AT)
```

```

    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists languages(
    lng_gui_AT char(10) not null,
    lng_language varchar(30) default 'unknown' not null,
    primary key (lng_gui_AT,lng_language),
    constraint guideLanguage foreign key (lng_gui_AT) references
guide(gui_AT)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists driver(
    drv_AT char(10) not null,
    drv_license enum('A','B','C','D') not null,
    drv_route enum('LOCAL','ABROAD') not null,
    drv_experience tinyint(4) not null,
    PRIMARY KEY(drv_AT),
    constraint driverWorkData foreign key (drv_AT) references worker(wrk_AT)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists admin(
    adm_AT char(10) not null,
    adm_type enum('LOGISTICS','ADMINISTRATIVE','ACCOUNTING') not null,
    adm_diploma varchar(200) DEFAULT 'unkown' not null,
    PRIMARY KEY(adm_AT),
    constraint workerAdminAt foreign key (adm_AT) references worker(wrk_AT)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists manages(
    mng_adm_AT char(10) default 'unkown' not null,
    mng_br_code int(11) not null ,
    primary key (mng_adm_AT, mng_br_code),
    constraint manageBranch foreign key (mng_br_code) references
branch(br_code)
    ON DELETE CASCADE ON UPDATE CASCADE,
    constraint managerAdministration foreign key (mng_adm_AT) references
admin(adm_AT)
    ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists trip(
    tr_id int(10) not null auto_increment,
    tr_departure datetime not null,
    tr_return datetime not null,
    tr_maxseats tinyint(4) not null,
    tr_cost float(7,2) not null,
    tr_br_code int(11) not null,
    tr_gui_AT char(10) not null,
    tr_drv_AT char(10) not null,
    PRIMARY KEY(tr_id),
    constraint tripBranchCode foreign key (tr_br_code) references
branch(br_code)
    ON DELETE CASCADE ON UPDATE CASCADE,

```

```

constraint tripGuideAt foreign key (tr_gui_AT) references guide(gui_AT)
ON DELETE CASCADE ON UPDATE CASCADE,
constraint tripDriverAt foreign key (tr_drv_AT) references driver(drv_AT)
ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists event(
    ev_tr_id int(11) not null,          #removed auto increment
    ev_start datetime not null,
    ev_end datetime not null,
    ev_descr text not null,
    primary key (ev_tr_id, ev_start),
    constraint tripEventId foreign key (ev_tr_id) references trip(tr_id)
ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists reservation(
    res_tr_id int(11) not null auto_increment,
    res_seatnum tinyint(4) not null,
    res_name varchar(20) default 'unknown' not null,
    res_lname varchar(20) default 'unknown' not null,
    res_isadult enum('ADULT','MINOR') not null,
    primary key (res_tr_id, res_seatnum),
    constraint tripReservationId foreign key (res_tr_id) references
trip(tr_id)
ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists travelTo(
    to_tr_id int(11) not null auto_increment,
    to_dst_id int(11) not null,
    to_arrival datetime not null,
    to_departure datetime not null,
    primary key (to_tr_id, to_dst_id),
    constraint travelIdTrip foreign key (to_tr_id) references trip(tr_id)
ON DELETE CASCADE ON UPDATE CASCADE,
    constraint destinationIdTrip foreign key (to_dst_id) references
destination(dst_id)
ON DELETE CASCADE ON UPDATE CASCADE
);

create table if not exists offers(
    offer_id int(11) not null auto_increment,
    offer_startDate date not null,
    offer_endDate date not null,
    cost_per_person float(7,2) not null,
    offer_dst_id int(11) not null,
    primary key (offer_id),
    constraint offerDestination foreign key (offer_dst_id) references
destination(dst_id)
on delete cascade on update cascade
);

create table if not exists reservation_offers(
    res_offer_id int(11) not null auto_increment,
    res_name varchar(20) not null,
    res_lname varchar(20) not null,

```

```

        offer_id_trip int(11) not null,
        res_deposit float(7,2) not null,
        primary key (res_offer_id),
        constraint offerId foreign key (offer_id_trip) references
offers(offer_id)
        on delete cascade on update cascade
    );

create table if not exists itOfficer(
    it_AT char(10) not null,
    password char(10) default 'password' not null,
    start_date date not null,
    end_date date,
    primary key (IT_AT),
    constraint itworker foreign key (IT_AT) references worker(wrk_AT)
    on delete cascade on update cascade
);

create table if not exists log(
    logid int not null auto_increment,
    logdescrc text not null,
    log_ID_AT char(10) not null,
    log_date datetime not null,
    primary key(logid),
    constraint logitat foreign key (log_ID_AT) references itOfficer(IT_AT)
    on delete cascade on update cascade
);

drop table if exists templogin;
create table templogin(
    tempID char(10) primary key
);

INSERT INTO branch VALUES
(1, 1, 'Main Street', 'New York'),
(2, 2, 'Maple Avenue', 'Chicago'),
(3, 3, '1st Street', 'Los Angeles'),
(4, 4, '2nd Avenue', 'Houston'),
(5, 5, '3rd Boulevard', 'Philadelphia'),
(6, 1, '4th Street', 'Phoenix'),
(7, 2, '5th Avenue', 'San Antonio'),
(8, 3, '6th Boulevard', 'San Diego'),
(9, 4, '7th Street', 'Dallas'),
(10, 5, '8th Avenue', 'San Jose');

INSERT INTO phones VALUES
(1, '0201111111'),
(2, '0202222222'),
(3, '0203333333'),
(4, '0204444444'),
(5, '0205555555'),
(6, '0206666666'),
(7, '0207777777'),
(8, '0208888888'),
(9, '0209999999'),
(10, '0200000000'),

```

```
(1, '0201234567'),
(2, '0202345678'),
(3, '0203456789'),
(4, '0204567890'),
(5, '0205678901'),
(6, '0206789012'),
(7, '0207890123'),
(8, '0208901234'),
(9, '0209012345'),
(10, '0200123456'),
(1, '0205432109'),
(2, '0204321098'),
(3, '0203210987'),
(4, '0202109876'),
(5, '0201098765'),
(6, '0200987654'),
(7, '0209876543'),
(8, '0208765432'),
(9, '0207654321'),
(10, '0206543210');
```

INSERT INTO destination VALUES

```
(1, 'New York', 'The city that never sleeps', 'LOCAL', 'English', NULL),
(2, 'Chicago', 'The Windy City', 'LOCAL', 'English', NULL),
(3, 'Los Angeles', 'The City of Angels', 'LOCAL', 'English', NULL),
(4, 'Houston', 'Space City', 'LOCAL', 'English', NULL),
(5, 'Philadelphia', 'The City of Brotherly Love', 'LOCAL', 'English', NULL),
(6, 'Phoenix', 'The Valley of the Sun', 'LOCAL', 'English', NULL),
(7, 'San Antonio', 'The Alamo City', 'LOCAL', 'English', NULL),
(8, 'San Diego', 'America's Finest City', 'LOCAL', 'English', NULL),
(9, 'Dallas', 'Big D', 'LOCAL', 'English', NULL),
(10, 'San Jose', 'The Capital of Silicon Valley', 'LOCAL', 'English', NULL),
(11, 'Athens', 'The Capital of Greece', 'LOCAL', 'Greek', 7),
(12, 'Thessaloniki', 'The Second City of Greece', 'LOCAL', 'Greek', 7),
(13, 'Rome', 'The Eternal City', 'ABROAD', 'Italian', 1),
(14, 'Paris', 'The City of Love', 'ABROAD', 'French', 2),
(15, 'London', 'The Capital of the UK', 'ABROAD', 'English', 3),
(16, 'Madrid', 'The Capital of Spain', 'ABROAD', 'Spanish', 4),
(17, 'Berlin', 'The Capital of Germany', 'ABROAD', 'German', 5),
(18, 'Tokyo', 'The Capital of Japan', 'ABROAD', 'Japanese', 6),
(19, 'Barcelona', 'The Capital of Catalonia', 'ABROAD', 'Spanish', NULL),
(20, 'Amsterdam', 'The Capital of the Netherlands', 'ABROAD', 'Dutch', NULL),
(21, 'Sydney', 'The Capital of Australia', 'ABROAD', 'English', NULL),
(22, 'Beijing', 'The Capital of China', 'ABROAD', 'Chinese', NULL),
(23, 'Moscow', 'The Capital of Russia', 'ABROAD', 'Russian', NULL),
(24, 'Dubai', 'The City of Gold', 'ABROAD', 'Arabic', NULL),
(25, 'Rio de Janeiro', 'The Marvelous City', 'ABROAD', 'Portuguese', NULL),
(26, 'Cape Town', 'The Mother City', 'ABROAD', 'English', NULL),
(27, 'Johannesburg', 'The City of Gold', 'ABROAD', 'Afrikaans', NULL),
(28, 'São Paulo', 'The Capital of São Paulo', 'ABROAD', 'Portuguese', NULL),
(29, 'Lisbon', 'The Capital of Portugal', 'ABROAD', 'Portuguese', NULL),
(30, 'Bangkok', 'The Capital of Thailand', 'ABROAD', 'Thai', NULL);
```

INSERT INTO worker VALUES

```
('AT000001', 'John', 'Doe', 45000.00, 1),
('AT000002', 'Jane', 'Doe', 50000.00, 2),
```

```
('AT000003', 'Bob', 'Smith', 40000.00, 3),
('AT000004', 'Sue', 'Smith', 55000.00, 4),
('AT000005', 'Alice', 'Jones', 52000.00, 5),
('AT000006', 'Mike', 'Jones', 48000.00, 6),
('AT000007', 'Sarah', 'Williams', 45000.00, 7),
('AT000008', 'Chris', 'Williams', 52000.00, 8),
('AT000009', 'Dave', 'Johnson', 50000.00, 9),
('AT000010', 'Emily', 'Johnson', 45000.00, 10),
('AT000011', 'Laura', 'Brown', 52000.00, 1),
('AT000012', 'Tom', 'Brown', 48000.00, 2),
('AT000013', 'Kim', 'Davis', 45000.00, 3),
('AT000014', 'Steve', 'Davis', 52000.00, 4),
('AT000015', 'Julie', 'Moore', 50000.00, 5),
('AT000016', 'Jason', 'Moore', 45000.00, 6),
('AT000017', 'Samantha', 'Miller', 52000.00, 7),
('AT000018', 'Mark', 'Miller', 48000.00, 8),
('AT000019', 'Alex', 'Wilson', 45000.00, 9),
('AT000020', 'Megan', 'Wilson', 52000.00, 10),
('AT000021', 'Tina', 'Taylor', 50000.00, 1),
('AT000022', 'Ryan', 'Taylor', 45000.00, 2),
('AT000023', 'Rachel', 'Anderson', 52000.00, 3),
('AT000024', 'Adam', 'Anderson', 48000.00, 4),
('AT000025', 'Katie', 'Thomas', 45000.00, 5),
('AT000026', 'Scott', 'Thomas', 52000.00, 6),
('AT000027', 'Amy', 'Jackson', 50000.00, 7),
('AT000028', 'Brian', 'Jackson', 45000.00, 8),
('AT000029', 'Lisa', 'White', 52000.00, 9),
('AT000030', 'Matt', 'White', 48000.00, 10),
('AT000031', 'George', 'Baknis', 12000.00, 1),
('AT000032', 'Thanasis', 'Bakalis', 12000.00, 1),
('AT000033', 'Haris', 'Stamelos', 12000.00, 1),
('AT000034', 'Maria', 'Papadopoulou', 46000.00, 1),
('AT000035', 'Nikos', 'Georgiou', 48000.00, 2),
('AT000036', 'Giannis', 'Papadopoulos', 50000.00, 3),
('AT000037', 'Eleni', 'Katsarou', 45000.00, 4),
('AT000038', 'Dimitris', 'Nikolopoulos', 52000.00, 5),
('AT000039', 'Christina', 'Papadaki', 55000.00, 6),
('AT000040', 'Panos', 'Kostopoulos', 42000.00, 7),
('AT000041', 'Andreas', 'Ioannou', 48000.00, 8),
('AT000042', 'Sofia', 'Christou', 45000.00, 9),
('AT000043', 'Katerina', 'Papageorgiou', 52000.00, 10),
('AT000044', 'Marios', 'Antoniou', 48000.00, 1),
('AT000045', 'Stavros', 'Papadopoulos', 45000.00, 2),
('AT000046', 'Kostas', 'Vasilopoulos', 52000.00, 3),
('AT000047', 'Eva', 'Karakosta', 50000.00, 4),
('AT000048', 'Maria', 'Karagianni', 45000.00, 5),
('AT000049', 'Dimitris', 'Christopoulos', 52000.00, 6),
('AT000050', 'Eleni', 'Kontou', 48000.00, 7),
('AT000051', 'Sofia', 'Papaioannou', 45000.00, 8),
('AT000052', 'Katerina', 'Papadimitriou', 52000.00, 9),
('AT000053', 'Panos', 'Stefanopoulos', 50000.00, 10),
('AT000054', 'Stavros', 'Nikolaidis', 45000.00, 1),
('AT000055', 'Kostas', 'Christodoulou', 52000.00, 2),
('AT000056', 'Eva', 'Vlachou', 48000.00, 3),
('AT000057', 'Maria', 'Papandreou', 45000.00, 4),
('AT000058', 'Dimitris', 'Andreadis', 52000.00, 5),
('AT000059', 'Eleni', 'Mavridou', 48000.00, 6),
```



```

('AT000060', 'Sofia', 'Papadopoulou', 45000.00, 7);

INSERT INTO guide VALUES
('AT000001', 'I have been working as a tour guide for 10 years and have
extensive knowledge of local history and landmarks.'),
('AT000002', 'I have a degree in art history and have been leading museum and
gallery tours for the past 5 years.'),
('AT000003', 'I am fluent in Spanish, French, and English and have been
leading international tours for the past 7 years.'),
('AT000004', 'I have a background in geology and have been leading hiking and
outdoor adventure tours for the past 3 years.'),
('AT000005', 'I have been leading food and wine tours in the local area for
the past 8 years and have a deep understanding of regional cuisine and
culinary traditions.'),
('AT000006', 'I have a degree in theatre and have been leading backstage and
behind-the-scenes tours at local performance venues for the past 5 years.'),
('AT000007', 'I am an experienced wildlife photographer and have been leading
photography tours in local nature reserves for the past 7 years.'),
('AT000008', 'I have a background in engineering and have been leading
technology and innovation tours for the past 3 years.'),
('AT000009', 'I am an expert in local architecture and have been leading
walking tours of historical neighborhoods for the past 8 years.'),
('AT000010', 'I have a degree in music and have been leading music and
entertainment tours for the past 5 years.');
```

```

INSERT INTO languages VALUES
('AT000001', 'Spanish'),
('AT000001', 'French'),
('AT000002', 'Greek'),
('AT000003', 'Italian'),
('AT000003', 'French'),
('AT000004', 'Spanish'),
('AT000005', 'German'),
('AT000006', 'French'),
('AT000007', 'Portuguese'),
('AT000008', 'Italian'),
('AT000009', 'Greek'),
('AT000010', 'French');
```

```

INSERT INTO driver VALUES
('AT000011', 'B', 'LOCAL', 5),
('AT000012', 'C', 'LOCAL', 10),
('AT000013', 'D', 'LOCAL', 15),
('AT000014', 'A', 'ABROAD', 5),
('AT000015', 'B', 'ABROAD', 10),
('AT000016', 'C', 'ABROAD', 15),
('AT000017', 'D', 'ABROAD', 5),
('AT000018', 'A', 'LOCAL', 10),
('AT000019', 'B', 'LOCAL', 15),
('AT000020', 'C', 'ABROAD', 5);

INSERT INTO admin VALUES
('AT000021', 'ADMINISTRATIVE', 'Bachelor's degree in Business
Administration'),
('AT000022', 'ADMINISTRATIVE', 'Bachelor's degree in Finance'),
('AT000023', 'ADMINISTRATIVE', 'Bachelor's degree in Public
```

```

Administration'),
('AT000024', 'ADMINISTRATIVE', 'Bachelor's degree in Supply Chain
Management'),
('AT000025', 'ADMINISTRATIVE', 'Bachelor's degree in Accounting'),
('AT000026', 'ADMINISTRATIVE', 'Bachelor's degree in Organizational
Management'),
('AT000027', 'ADMINISTRATIVE', 'Bachelor's degree in Logistics and
Transportation'),
('AT000028', 'ADMINISTRATIVE', 'Bachelor's degree in Financial Management'),
('AT000029', 'ADMINISTRATIVE', 'Bachelor's degree in Business
Administration'),
('AT000030', 'ADMINISTRATIVE', 'Bachelor's degree in Industrial
Management'),
('AT000031', 'ACCOUNTING', 'Bachelor's degree in Health Services
Administration'),
('AT000032', 'ACCOUNTING', 'Bachelor's degree in Criminal Justice
Administration'),
('AT000033', 'ACCOUNTING', 'Bachelor's degree in Education Administration'),
('AT000034', 'ACCOUNTING', 'Bachelor's degree in Hospitality
Administration'),
('AT000035', 'ACCOUNTING', 'Bachelor's degree in Environmental Policy and
Management'),
('AT000036', 'ACCOUNTING', 'Bachelor's degree in International Business'),
('AT000037', 'ACCOUNTING', 'Bachelor's degree in Public Policy'),
('AT000038', 'ACCOUNTING', 'Bachelor's degree in Project Management'),
('AT000039', 'ACCOUNTING', 'Bachelor's degree in Social Work
Administration'),
('AT000040', 'ACCOUNTING', 'Bachelor's degree in Nonprofit Administration'),
('AT000041', 'LOGISTICS', 'Bachelor's degree in Communication Studies'),
('AT000042', 'LOGISTICS', 'Bachelor's degree in Information Technology
Management'),
('AT000043', 'LOGISTICS', 'Bachelor's degree in Marketing Administration'),
('AT000044', 'LOGISTICS', 'Bachelor's degree in Retail Management'),
('AT000045', 'LOGISTICS', 'Bachelor's degree in Sports Administration'),
('AT000046', 'LOGISTICS', 'Bachelor's degree in Public Safety
Administration'),
('AT000047', 'LOGISTICS', 'Bachelor's degree in Emergency Management'),
('AT000048', 'LOGISTICS', 'Bachelor's degree in Emergency Management'),
('AT000049', 'LOGISTICS', 'Bachelor's degree in Emergency Management'),
('AT000050', 'LOGISTICS', 'Bachelor's degree in Emergency Management');

```

```

INSERT INTO manages VALUES

```

```

('AT000021', 1),
('AT000022', 2),
('AT000023', 3),
('AT000024', 4),
('AT000025', 5),
('AT000026', 6),
('AT000027', 7),
('AT000028', 8),
('AT000029', 9),
('AT000030', 10),
('AT000031', 1),
('AT000032', 2),
('AT000033', 3),

```

```
('AT000034', 4),
('AT000035', 5),
('AT000036', 6),
('AT000037', 7),
('AT000038', 8),
('AT000039', 9),
('AT000040', 10),
('AT000041', 1),
('AT000042', 2),
('AT000043', 3),
('AT000044', 4),
('AT000045', 5),
('AT000046', 6),
('AT000047', 7),
('AT000048', 8),
('AT000049', 9),
('AT000050', 10);
```

INSERT INTO trip VALUES

```
(1, '2022-01-01', '2022-01-07', 30, 1000.00, 1, 'AT000001', 'AT000011'),
(2, '2022-01-08', '2022-01-14', 35, 1200.00, 2, 'AT000002', 'AT000012'),
(3, '2022-01-15', '2022-01-21', 40, 1500.00, 3, 'AT000003', 'AT000013'),
(4, '2022-01-22', '2022-01-28', 45, 1700.00, 4, 'AT000004', 'AT000014'),
(5, '2022-01-29', '2022-02-04', 50, 2000.00, 5, 'AT000005', 'AT000015'),
(6, '2022-02-05', '2022-02-11', 55, 2200.00, 6, 'AT000006', 'AT000016'),
(7, '2022-02-12', '2022-02-18', 60, 2500.00, 7, 'AT000007', 'AT000017'),
(8, '2022-02-19', '2022-02-25', 65, 2700.00, 8, 'AT000008', 'AT000018'),
(9, '2022-02-26', '2022-03-04', 70, 3000.00, 9, 'AT000009', 'AT000019'),
(10, '2022-03-05', '2022-03-11', 75, 3200.00, 10, 'AT000010', 'AT000020'),
(11, '2022-03-12', '2022-03-18', 80, 3500.00, 1, 'AT000001', 'AT000011'),
(12, '2022-03-19', '2022-03-25', 85, 3700.00, 2, 'AT000002', 'AT000012'),
(13, '2022-03-26', '2022-04-01', 90, 4000.00, 3, 'AT000003', 'AT000013'),
(14, '2022-04-02', '2022-04-08', 95, 4200.00, 4, 'AT000004', 'AT000014'),
(15, '2022-04-09', '2022-04-15', 100, 4500.00, 5, 'AT000005', 'AT000015'),
(16, '2022-04-16', '2022-04-22', 105, 4700.00, 6, 'AT000006', 'AT000016'),
(17, '2022-04-23', '2022-04-29', 110, 5000.00, 7, 'AT000007', 'AT000017'),
(18, '2022-04-30', '2022-05-06', 115, 5200.00, 8, 'AT000008', 'AT000018'),
(19, '2022-05-07', '2022-05-13', 20, 1200.00, 9, 'AT000009', 'AT000019'),
(20, '2022-05-14', '2022-05-20', 25, 1100.00, 10, 'AT000010', 'AT000020'),
(21, '2022-05-21', '2022-05-27', 30, 1000.00, 1, 'AT000001', 'AT000011'),
(22, '2022-05-28', '2022-06-03', 35, 900.00, 2, 'AT000002', 'AT000012'),
(23, '2022-06-04', '2022-06-10', 40, 800.00, 3, 'AT000003', 'AT000013'),
(24, '2022-06-11', '2022-06-17', 45, 700.00, 4, 'AT000004', 'AT000014'),
(25, '2022-06-18', '2022-06-24', 50, 600.00, 5, 'AT000005', 'AT000015'),
(26, '2022-06-25', '2022-07-01', 55, 500.00, 6, 'AT000006', 'AT000016'),
(27, '2022-07-02', '2022-07-08', 60, 400.00, 7, 'AT000007', 'AT000017'),
(28, '2022-07-09', '2022-07-15', 65, 300.00, 8, 'AT000008', 'AT000018'),
(29, '2022-07-16', '2022-07-22', 70, 200.00, 9, 'AT000009', 'AT000019'),
(30, '2022-07-23', '2022-07-29', 75, 100.00, 10, 'AT000010', 'AT000020');
```

INSERT INTO event VALUES

```
(1, '2022-01-01', '2022-01-07', 'Event 1'),
(1, '2022-01-02', '2022-01-06', 'Event 1.1'),
(2, '2022-01-08', '2022-01-14', 'Event 2'),
(2, '2022-01-10', '2022-01-11', 'Event 2.2'),
(3, '2022-01-15', '2022-01-21', 'Event 3');
```

```

(4, '2022-01-22', '2022-01-28', 'Event 4'),
(5, '2022-01-29', '2022-02-04', 'Event 5'),
(6, '2022-02-05', '2022-02-11', 'Event 6'),
(7, '2022-02-12', '2022-02-18', 'Event 7'),
(8, '2022-02-19', '2022-02-25', 'Event 8'),
(9, '2022-02-26', '2022-03-04', 'Event 9'),
(10, '2022-03-05', '2022-03-11', 'Event 10'),
(11, '2022-03-12', '2022-03-18', 'Event 11'),
(12, '2022-03-19', '2022-03-25', 'Event 12'),
(13, '2022-03-26', '2022-04-01', 'Event 13'),
(14, '2022-04-02', '2022-04-08', 'Event 14'),
(15, '2022-04-09', '2022-04-15', 'Event 15'),
(16, '2022-04-16', '2022-04-22', 'Event 16'),
(17, '2022-04-23', '2022-04-29', 'Event 17'),
(18, '2022-04-30', '2022-05-06', 'Event 18'),
(19, '2022-05-07', '2022-05-13', 'Event 19'),
(20, '2022-05-14', '2022-05-20', 'Event 20'),
(21, '2022-05-21', '2022-05-27', 'Event 21'),
(22, '2022-05-28', '2022-06-03', 'Event 22'),
(23, '2022-06-04', '2022-06-10', 'Event 23'),
(24, '2022-06-11', '2022-06-17', 'Event 24'),
(25, '2022-06-18', '2022-06-24', 'Event 25'),
(26, '2022-06-25', '2022-07-01', 'Event 26'),
(27, '2022-07-02', '2022-07-08', 'Event 27'),
(28, '2022-07-09', '2022-07-15', 'Event 28'),
(29, '2022-07-16', '2022-07-22', 'Event 29'),
(30, '2022-07-23', '2022-07-29', 'Event 30');

```

INSERT INTO reservation VALUES

```

(1, 1, 'John', 'Doe', 'ADULT'),
(1, 2, 'Jane', 'Doe', 'ADULT'),
(1, 3, 'Tom', 'Smith', 'MINOR'),
(2, 1, 'Alice', 'Johnson', 'ADULT'),
(2, 2, 'Bob', 'Johnson', 'MINOR'),
(3, 1, 'Emily', 'Williams', 'ADULT'),
(3, 2, 'William', 'Williams', 'ADULT'),
(4, 1, 'Samantha', 'Jones', 'ADULT'),
(4, 2, 'Jason', 'Jones', 'MINOR'),
(5, 1, 'Emma', 'Brown', 'ADULT'),
(5, 2, 'David', 'Brown', 'MINOR'),
(6, 1, 'Olivia', 'Davis', 'ADULT'),
(6, 2, 'James', 'Davis', 'MINOR'),
(7, 1, 'Ava', 'Miller', 'ADULT'),
(7, 2, 'Benjamin', 'Miller', 'MINOR'),
(8, 1, 'Isabella', 'Moore', 'ADULT'),
(8, 2, 'Michael', 'Moore', 'MINOR'),
(9, 1, 'Mia', 'Taylor', 'ADULT'),
(9, 2, 'Matthew', 'Taylor', 'MINOR'),
(10, 1, 'Abigail', 'Anderson', 'ADULT'),
(10, 2, 'Christopher', 'Anderson', 'MINOR'),
(11, 10, 'Emily', 'Thomas', 'ADULT'),
(1, 11, 'John', 'Doe', 'ADULT'),
(1, 21, 'Jane', 'Doe', 'ADULT'),
(1, 31, 'Bill', 'Smith', 'ADULT'),
(1, 41, 'Susan', 'Smith', 'ADULT'),
(1, 51, 'Paul', 'Johnson', 'MINOR'),

```

```

(2, 11, 'Emily', 'Johnson', 'MINOR'),
(2, 21, 'Michael', 'Williams', 'ADULT'),
(2, 31, 'Samantha', 'Williams', 'ADULT'),
(2, 41, 'Roger', 'Brown', 'ADULT');

INSERT INTO travelTo VALUES
(1, 1, '2022-01-07', '2022-01-01'),
(2, 2, '2022-01-14', '2022-01-08'),
(3, 3, '2022-01-21', '2022-01-15'),
(4, 4, '2022-01-28', '2022-01-22'),
(5, 5, '2022-02-04', '2022-01-29'),
(6, 6, '2022-02-11', '2022-02-05'),
(7, 7, '2022-02-18', '2022-02-12'),
(8, 8, '2022-02-25', '2022-02-19'),
(9, 9, '2022-03-04', '2022-02-26'),
(10, 10, '2022-03-11', '2022-03-05'),
(11, 11, '2022-03-18', '2022-03-12'),
(12, 12, '2022-03-25', '2022-03-19'),
(13, 13, '2022-04-01', '2022-03-26'),
(14, 14, '2022-04-08', '2022-04-02'),
(15, 15, '2022-04-15', '2022-04-09'),
(16, 16, '2022-04-22', '2022-04-16'),
(17, 17, '2022-04-29', '2022-04-23'),
(18, 18, '2022-05-06', '2022-04-30'),
(19, 19, '2022-05-13', '2022-05-07'),
(20, 20, '2022-05-20', '2022-05-14'),
(21, 21, '2022-05-27', '2022-05-21'),
(22, 22, '2022-06-03', '2022-05-28'),
(23, 23, '2022-06-10', '2022-06-04'),
(24, 24, '2022-06-17', '2022-06-11'),
(25, 25, '2022-06-24', '2022-06-18'),
(26, 26, '2022-07-01', '2022-06-25'),
(27, 27, '2022-07-08', '2022-07-02'),
(28, 28, '2022-07-15', '2022-07-09'),
(29, 29, '2022-07-22', '2022-07-16'),
(30, 30, '2022-07-29', '2022-07-23');

insert into itOfficer values('AT000051', '1234', NOW(), NULL);
insert into itOfficer values('AT000052', '1234', NOW(), NULL);
insert into itOfficer values('AT000053', '1234', NOW(), NULL);

insert into offers values
(null, '2023-01-01', '2024-01-01', 250, 1),
(null, '2023-01-10', '2024-02-01', 300, 2),
(null, '2023-05-01', '2024-05-25', 400, 3);

#Procedures

#FIRST PROCEDURE
drop procedure if exists newDriverAssignment;
DELIMITER $

CREATE PROCEDURE newDriverAssignment(drvAT char(10), drvName varchar(20),
drvLname varchar(30),
drvSalary float(7,2), drvLicense enum('A', 'B', 'C', 'D'), drvRoute
enum('LOCAL', 'ABROAD'), drvExperience tinyint(4))

```

```

BEGIN
    declare BRANCH_CODE int;

    select br_code into BRANCH_CODE
    from branch
    inner join worker on branch.br_code = worker.wrk_br_code
    inner join driver on worker.wrk_AT = driver.driv_AT
    group by br_code
    order by count(driv_AT) asc limit 1;

    insert into worker values(drvAT, drvName, drvLname, drvSalary,
BRANCH_CODE);
    insert into driver values(drvAT, drvLicense, drvRoute, drvExperience);

END$

delimiter ;

#SECOND PROCEDURE
drop procedure if exists showAvailableTrips;
delimiter $

create procedure showAvailableTrips(brCode int(11), startDate date, endDate
date)
begin
    declare not_found int;
    declare tripId int(11);
    declare tripCost float(7,2);
    declare maxSeats tinyint(4);
    declare reservCount tinyint(4);
    declare emptySeats tinyint(4);
    declare drivLname varchar(20);
    declare drivName varchar(20);
    declare GuideLname varchar(20);
    declare GuideName varchar(20);
    declare guideAT char(10);
    declare driverAt char(10);
    declare departureDate datetime;
    declare returnDate datetime;

    declare tcursor cursor for
    select tr_id, tr_departure, tr_return, tr_maxseats, tr_cost, tr_gui_AT,
tr_driv_AT
    from trip
    where tr_br_code = brCode and tr_departure >= startDate and tr_departure
<= endDate;

    declare continue handler for not found set not_found=1;

    set not_found=0;
    open tcursor;

    repeat
        fetch tcursor into tripId, departureDate, returnDate, maxSeats,
tripCost, guideAT, driverAt;
        if(not_found=0) then
            select count(res_tr_id) into reservCount from reservation where

```

```

res_tr_id = tripId;
    set emptySeats = maxSeats - reservCount;
    select wrk_lame, wrk_name into GuideLname, GuideName from worker
where wrk_AT = guideAT;
    select wrk_lame, wrk_name into drivLname, drivName from worker
where wrk_AT = driverAt;

    select tripCost, maxSeats, reservCount, emptySeats, drivLname,
    drivName, GuideLname, GuideName, departureDate, returnDate;
    end if;
until (not_found=1)
end repeat;

close tcursor;
end $

DELIMITER ;

#THIRD PROCEDURE
drop procedure if exists delete_admin;
DELIMITER $

CREATE PROCEDURE delete_admin(IN first_name VARCHAR(20), IN last_name
VARCHAR(20))
BEGIN
    -- check if employee is a manager and his/her admin type.
    SELECT COUNT(*) INTO @is_manager
    FROM manages
    INNER JOIN worker ON manages.mng_adm_AT = worker.wrk_AT
    WHERE worker.wrk_name = first_name AND worker.wrk_lame = last_name;

    -- If the employee is a branch manager
    IF @is_manager > 0 THEN
        SELECT wrk_AT INTO @AT
        from worker
        INNER JOIN manages on manages.mng_adm_AT = wrk_AT
        WHERE worker.wrk_name = first_name AND worker.wrk_lame = last_name;

        -- And if he/she is of type ADMINISTRATIVE
        SELECT count(*) INTO @is_administrative
        from admin
        where admin.adm_AT = @AT AND admin.adm_type = 'ADMINISTRATIVE';

        -- Cannot be deleted
        IF @is_administrative > 0 THEN
            signal sqlstate value '45000'
            set message_text = 'Cannot delete administrator of a branch';
        -- Else delete
        ELSE
            DELETE admin FROM admin
            INNER JOIN worker ON admin.adm_AT = worker.wrk_AT
            WHERE worker.wrk_name = first_name AND worker.wrk_lame =
last_name;
            select concat(first_name, ' ', last_name, ' ', 'with AT: ', @AT,
' deleted successfully');
            #select 'not null';
        END IF;
    END IF;

```

```

-- Employee is not a manager
ELSE
    select concat(first_name, ' ', last_name, ' ', 'with AT: ', @AT, 'is
not a manager.');
```

```

    END IF;
END$

delimiter ;

#PROCEDURES 4.1 4.2
#drop index idx on reservation_offers;
CREATE INDEX idx ON reservation_offers (res_deposit, res_lname);

drop procedure if exists deposit_range_names;
delimiter $

create procedure deposit_range_names(IN minDeposit INT , IN maxDeposit INT)
BEGIN
    select res_name,res_lname from reservation_offers where
res_deposit>=minDeposit and res_deposit <=maxDeposit;
end $
delimiter ;

drop procedure if exists data_last_name;
delimiter $

create procedure data_last_name(IN lastName VARCHAR(20))
BEGIN
    select res_name, res_lname,offers.* from offers inner join
reservation_offers on offers.offer_id = reservation_offers.offer_id_trip
where reservation_offers.res_lname = lastName;

    if found_rows()>1 then
        select offers.*,count(res_lname) from offers
        inner join reservation_offers on offers.offer_id =
reservation_offers.offer_id_trip
        where reservation_offers.res_lname = lastName
        group by offer_id;
    end if;
end $
delimiter ;

#procedure to get id for triggers
drop procedure if exists getItID;
delimiter $
create procedure getItID(in insertedItId char(10), in insertedItPass
char(10), out itID char(10))
begin
    declare verificationValue char(10);

    select it_AT into verificationValue from itOfficer where it_AT =
insertedItId and password = insertedItPass;
    if (verificationValue is not null) then
        select insertedItId into itID;
        insert into templogin values(insertedItId);
    end if;
end if;

```



```

end $

#TRIGGRES

#first set of triggers
drop trigger if exists tripInsertion;
delimiter $
create trigger tripInsertion after insert on trip
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Inserted a trip',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists tripUpdating;
delimiter $
create trigger tripUpdating after update on trip
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Updated a trip.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists tripDeletion;
delimiter $
create trigger tripDeletion after delete on trip
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Deleted a trip.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $

```

```

end $
delimiter ;

drop trigger if exists reservationInsertion;
delimiter $
create trigger reservationInsertion after insert on reservation
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Inserted a
reservation.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists reservationUpdating;
delimiter $
create trigger reservationUpdating after update on reservation
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Updated a
reservation.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists reservationDeletion;
delimiter $
create trigger reservationDeletion after delete on reservation
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Deleted a
reservation.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $

```

```

end $
delimiter ;

drop trigger if exists eventInsertion;
delimiter $
create trigger eventInsertion after insert on event
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
            set logdescrc = 'Inserted an
event.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists eventUpdating;
delimiter $
create trigger eventUpdating after update on event
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
            set logdescrc = 'Updated an
event.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;

end $
delimiter ;

drop trigger if exists eventDeletion;
delimiter $
create trigger eventDeletion after delete on event
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
            set logdescrc = 'Deleted an
event.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;

end $

```

```

delimiter ;

drop trigger if exists traveltoInsertion;
delimiter $
create trigger traveltoInsertion after insert on travelTo
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Inserted a
travelto.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists traveltoUpdating;
delimiter $
create trigger traveltoUpdating after update on travelTo
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Updated a
travelto.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;

end $
delimiter ;

drop trigger if exists traveltoDeletion;
delimiter $
create trigger traveltoDeletion after delete on travelTo
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Deleted a
travelto.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;

end $
delimiter ;

```

```

drop trigger if exists destinationInsertion;
delimiter $
create trigger destinationInsertion after insert on destination
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Inserted a
destination.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;
end $
delimiter ;

drop trigger if exists destinationUpdating;
delimiter $
create trigger destinationUpdating after update on destination
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Updated a
destination.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;

end $
delimiter ;

drop trigger if exists destinationDeletion;
delimiter $
create trigger destinationDeletion after delete on destination
for each row
begin
    declare testvalue char(10);
    select tempID into testvalue from templogin;
    if(testvalue is not null) then
        insert into log
        set logdescrc = 'Deleted a
destination.',log_ID_AT=testvalue,log_date=now();
    else
        signal sqlstate value '45000'
        set message_text = 'Error!You have to login as It Officer.';
    end if;

end $
delimiter ;

```

```

#second trigger
drop trigger if exists changeDateAndCost;

delimiter $

create trigger changeDateAndCost before update on trip
for each row
begin
    declare tempSeatsValue int;

    select count(*) into tempSeatsValue
    from trip
    inner join reservation r on trip.tr_id = r.res_tr_id
    where tr_id=new.tr_id
    group by res_tr_id;

    if(tempSeatsValue>0) then
        if(new.tr_cost <> old.tr_cost) then
            signal sqlstate value '45000'
            set message_text = 'You can not change the cost for this trip';
        end if;
        if(new.tr_departure <> old.tr_departure) then
            signal sqlstate value '45000'
            set message_text = 'You can not change the departure date for
this trip';
        end if;
        if(new.tr_return <> old.tr_return) then
            signal sqlstate value '45000'
            set message_text = 'You can not change the return date for this
trip';
        end if;
    end if;
end $
delimiter ;

#third trigger
drop trigger if exists changeWorkerSalary;

delimiter $

create trigger changeWorkerSalary before update on worker
for each row
begin
    if(new.wrk_salary<old.wrk_salary) then
        signal sqlstate value '45000'
        set message_text = 'There can be no reduction in the workers salary';
    end if;
end $
DELIMITER ;

drop procedure if exists showBranchDetails;

delimiter $
CREATE PROCEDURE showBranchDetails(brCode INT)
BEGIN

```

```

declare not_found int;
declare temp_tr_id int;
declare temp_cost float(7, 2);
declare total_cost float(7, 2);

-- total income from the reservations
declare trip_cost_cursor cursor for
select tr_cost, tr_id
from trip
where tr_br_code = brCode;

declare continue handler for not found set not_found=1;

set not_found=0;
set total_cost=0.0;
open trip_cost_cursor;

repeat
    fetch trip_cost_cursor INTO temp_cost, temp_tr_id;
    if(not_found=0) then
        select count(res_tr_id) * temp_cost
        from reservation
        where res_tr_id = temp_tr_id INTO temp_cost;

        set total_cost = total_cost + temp_cost;
    end if ;
until (not_found=1)
end repeat ;
close trip_cost_cursor;

-- total number of reservations
select COUNT(ALL res_tr_id) INTO @numOfRes
from reservation
inner join trip t on reservation.res_tr_id = t.tr_id
where tr_br_code = brCode;

SELECT 'Branch Details' AS 'Result Type', br_code AS 'Code', br_num AS
'Number', br_street AS 'Street', br_city AS
'City', wrk_name AS 'admin (FN)', wrk_lame AS 'Last name (LN)',
@numOfRes AS 'Num of reservations', total_cost AS
'Income'
FROM branch
inner join worker w on branch.br_code = w.wrk_br_code
inner join admin a on w.wrk_AT = a.adm_AT
WHERE brCode = br_code and adm_type = 'ADMINISTRATIVE';
END$

delimiter ;

drop procedure if exists branchInfo;

delimiter $
CREATE PROCEDURE branchInfo (brCode INT)
BEGIN
    SELECT wrk_name AS 'First Name', wrk_lame AS 'Last Name', wrk_salary AS

```

```

'Salary'
FROM worker
WHERE wrk_br_code = brCode;

SELECT 'Total Salaries' AS 'Result Type', SUM(wrk_salary)
FROM worker
WHERE wrk_br_code = brCode;
END$

delimiter ;

drop procedure if exists addIT;
delimiter $
CREATE PROCEDURE addIT(in wrkAT char(10), in wrkName varchar(20), in wrkLname
varchar(20), in wrkSalary float(7, 2), in
    wrkBrCode int, in password char(10), in startD datetime, in endD
datetime)
BEGIN
    -- First insert a worker
    insert into worker values(wrkAT, wrkName, wrkLname, wrkSalary,
wrkBrCode);

    -- After insert on itofficer table
    insert into itOfficer values (wrkAT, password, startD, endD);

END $
delimiter ;

drop procedure if exists showLog;

delimiter $
CREATE PROCEDURE showLog()
BEGIN
    DECLARE not_found INT;
    DECLARE logD TEXT;
    DECLARE logID CHAR(10);
    DECLARE itLname VARCHAR(20);
    DECLARE logDate DATETIME;

    DECLARE logcursor CURSOR FOR
    SELECT logdescrc, log_ID_AT, log_date
    FROM log;

    DECLARE CONTINUE HANDLER FOR NOT FOUND SET not_found = 1;

    SET not_found = 0;
    OPEN logcursor;

    -- Create a temporary table to store the data
    CREATE TEMPORARY TABLE temp_log_result (
        log_desc TEXT,
        it_lastname VARCHAR(20),
        log_date DATETIME
    );

    FETCH_LOOP: LOOP

```



```

        FETCH logcursor INTO logD, logID, logDate;
        IF not_found = 1 THEN
            LEAVE FETCH_LOOP;
        END IF;

        SELECT wrk_lame INTO itLname
        FROM worker
        INNER JOIN itOfficer ON worker.wrk_AT = itOfficer.it_AT
        WHERE wrk_AT = logID;

        INSERT INTO temp_log_result (log_desc, it_lastname, log_date)
        VALUES (logD, itLname, logDate);
    END LOOP FETCH_LOOP;

    CLOSE logcursor;

    -- Fetch the complete result set from the temporary table
    SELECT log_desc, it_lastname, log_date FROM temp_log_result;

    -- Drop the temporary table
    DROP TEMPORARY TABLE temp_log_result;
END;

delimiter ;

-- Extra functionalities
drop procedure if exists branch_stats;

delimiter $
CREATE PROCEDURE branch_stats(brCode INT)
BEGIN
    declare not_found int;
    declare temp_tr_id int;
    declare temp_cost float(7, 2);
    declare total_cost float(7, 2);
    declare total_salaries float(13, 2);

    -- total income from the reservations
    declare trip_cost_cursor cursor for
    select tr_cost, tr_id
    from trip
    where tr_br_code = brCode;

    declare continue handler for not found set not_found=1;

    set not_found=0;
    set total_cost=0.0;
    open trip_cost_cursor;

    repeat
        fetch trip_cost_cursor INTO temp_cost, temp_tr_id;
        if(not_found=0) then
            select count(res_tr_id) * temp_cost
            from reservation
            where res_tr_id = temp_tr_id INTO temp_cost;

```

```

        set total_cost = total_cost + temp_cost;
    end if ;
    until (not_found=1)
end repeat ;
close trip_cost_cursor;

-- expenses = salaries
SELECT SUM(wrk_salary)
FROM worker
WHERE wrk_br_code = brCode INTO total_salaries;

select total_cost AS 'Income', total_salaries AS 'expenses', total_cost -
total_salaries AS 'Turnover';

END $

drop procedure if exists driver_card;

delimiter $
CREATE PROCEDURE driver_card(id char(10))
BEGIN
    -- id
    select wrk_name AS "First Name", wrk_lame AS "Last Name", wrk_AT AS "AT",
wrk_br_code AS 'Branch Code', wrk_salary AS
    "Salary", drv_license AS "License", drv_route AS "Route",
drv_experience AS "Experience"
    from worker
    inner join driver on drv_AT = worker.wrk_AT
    where wrk_AT = id;

    -- assigned trips
    select tr_id AS "Trip Id", tr_departure AS "departure", tr_return AS
"Return"
    from trip
    where tr_drv_AT = id;

END $

drop procedure if exists guide_card;

delimiter $
CREATE PROCEDURE guide_card(id char(10))
BEGIN
    -- id
    select wrk_name AS "First Name", wrk_lame AS "Last Name", wrk_AT AS "AT",
wrk_br_code AS 'Branch Code', wrk_salary AS
    "Salary"
    from worker
    inner join guide on guide.gui_AT = worker.wrk_AT
    where wrk_AT = id;

    -- languages
    select lng_language
    from languages

```

```

        where lng_gui_AT = id;

        -- assigned trips
        select tr_id AS "Trip Id", tr_departure AS "departure", tr_return AS
"Return"
        from trip
        where tr_gui_AT = id;

        -- guide cv
        select gui_cv AS "CV"
        from guide
        where gui_AT = id;

END $

drop procedure if exists admin_card;
delimiter $
CREATE PROCEDURE admin_card(id char(10))
BEGIN
    -- id
    select wrk_name AS "First Name", wrk_lame AS "Last Name", wrk_AT AS "AT",
wrk_br_code AS 'Branch Code', wrk_salary AS
    "Salary", adm_type AS "Type", adm_diploma AS 'Diploma'
    from worker
    inner join admin on adm_AT = wrk_AT
    where wrk_AT = id;

END $

```

7 Κώδικας Java

7.1 ConnectionToDb

```
package com.mycompany.travelagency;

import java.sql.*;

public class ConnectionToDb {
    private static final String URL =
"jdbc:mysql://localhost:3306/travelagency";
    private static final String USERNAME = "root";
    private static final String PASSWORD = "password";
    //Connection con = null;

    public static Connection connectMySQLDB() {
        try {

            return DriverManager.getConnection(URL, USERNAME, PASSWORD);

        } catch (SQLException ex) {
            System.out.println(ex);
            return null;
        }
    }
}
```

7.2 TravelAgency

```
package com.mycompany.travelagency;

public class TravelAgency {

    public static void main(String[] args) {
        Login app = new Login();
        app.setVisible(true);
    }
}
```

7.3 Login

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to
edit this template
 */
package com.mycompany.travelagency;

import java.sql.Connection;
import javax.swing.JFrame;
```

```

import javax.swing.JOptionPane;
import java.awt.Toolkit;
import java.awt.Dimension;
import java.sql.CallableStatement;
import java.sql.SQLException;
import java.sql.Types;

public class Login extends javax.swing.JFrame {

    /**
     * Creates new form Login
     */
    public Login() {
        initComponents();
        Toolkit toolKit = getToolkit();
        Dimension size = toolKit.getScreenSize();
        setLocation(size.width/2 - getWidth()/2, size.height/2 -
getHeight()/2);

    }

    /**
     * This method is called from within the constructor to initialize the
form.
     * WARNING: Do NOT modify this code. The content of this method is always
regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-
BEGIN: initComponents
    private void initComponents() {

        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jPasswordField1 = new javax.swing.JPasswordField();
        jButton1 = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setResizable(false);

        jLabel1.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N
        jLabel1.setText("Travel Agency Login");

        jLabel2.setFont(new java.awt.Font("Cantarell", 1, 15)); // NOI18N
        jLabel2.setText("Username");

        jLabel3.setFont(new java.awt.Font("Cantarell", 1, 15)); // NOI18N
        jLabel3.setText("Password");

        jTextField1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jTextField1ActionPerformed(evt);
            }
        });
    }
}

```

```

jPasswordField1.addActionListener(new java.awt.event.ActionListener()
{
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jPasswordField1ActionPerformed(evt);
    }
});

jButton1.setText("Login");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .add(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .add(layout.createSequentialGroup()
                    .addGap(107, 107, 107)
                    .addComponent(jLabel1)
                    .addGap(0, 0, Short.MAX_VALUE))
                .add(layout.createSequentialGroup()
                    .addGap(105, 105, 105))
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel3)
                .addComponent(jLabel2))
            .addGap(18, 18, 18))
        .add(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jTextField1)
            .addComponent(jPasswordField1)))
        .addGap(108, 108, 108))
    .add(layout.createSequentialGroup()
        .addGap(144, 144, 144)
        .addComponent(jButton1,
javax.swing.GroupLayout.PREFERRED_SIZE, 125,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addGap(43, 43, 43)
        .addComponent(jLabel1)

```

```

        .addGap(36, 36, 36)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel2)
        .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel3)
        .addComponent(jPasswordField1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(44, 44, 44)
        .addComponent(jButton1)
        .addContainerGap(67, Short.MAX_VALUE))
    );

    pack();
} // </editor-fold> // GEN-END: initComponents

private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_jTextField1ActionPerformed
    // TODO add your handling code here:
} // GEN-LAST:event_jTextField1ActionPerformed

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_jButton1ActionPerformed
    // TODO add your handling code here:
    try {
        // Establish connection to the database
        Connection con = ConnectionToDb.connectMySqlDB();
        CallableStatement cStmt = con.prepareCall("{call getItID(?, ?,
?)}");

        cStmt.setString(1, jTextField1.getText());
        char[] pass = jPasswordField1.getPassword();
        String password = new String(pass);
        cStmt.setString(2, password);
        cStmt.registerOutParameter(3, Types.VARCHAR);
        cStmt.execute();

        String loginResult = cStmt.getString(3);
        // Check if itOfficer or not and proceed accordingly
        if (loginResult != null) {
            // edw kanei switch to panel
            Menu m = new Menu();
            m.setTitle("IT: " + loginResult);
            m.setVisible(true);
            dispose();
        } else {
            // edw petaei error message kai stamataei
            JFrame f;
            f = new JFrame();
            JOptionPane.showMessageDialog(f, "Incorrect username or

```

```

password!", "Error", JOptionPane.ERROR_MESSAGE);
    }
    //Close connection
    con.close();

    }catch (SQLException a){
        System.out.print(a);
    }
}
} //GEN-LAST:event_jButton1ActionPerformed

private void jPasswordField1ActionPerformed(java.awt.event.ActionEvent
evt) { //GEN-FIRST:event_jPasswordField1ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jPasswordField1ActionPerformed

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting
code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {
        java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.loggi
ng.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {
        java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.loggi
ng.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {
        java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.loggi
ng.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
        java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.loggi
ng.Level.SEVERE, null, ex);
    }
}
//</editor-fold>

/*jdbc connection*/

```



```

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Login().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton jButton1;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JPasswordField jPasswordField1;
    private javax.swing.JTextField jTextField1;
    // End of variables declaration//GEN-END:variables
}

```

7.4 Menu

```

/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
 * default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to
 * edit this template
 */
package com.mycompany.travelagency;
import java.awt.Dimension;
import java.awt.Toolkit;
import java.sql.CallableStatement;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;
import javax.swing.JFrame;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
import java.sql.SQLException;

public class Menu extends javax.swing.JFrame {

    /**
     * Creates new form Menu
     */
    public Menu() {
        initComponents();
        Toolkit toolKit = getToolkit();
        Dimension size = toolKit.getScreenSize();
        setLocation(size.width/2 - getWidth()/2, size.height/2 -
getHeight()/2);

        //here sonoo is database name, root is username and password
        try {

```

```

        Connection con = ConnectionToDb.connectMySqlDB();
        /*Branch*/
        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from branch");
        DefaultTableModel tbModel =
(DefaultTableModel) jTable2.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String br_code = String.valueOf(rs.getInt("br_code"));
            String br_num = String.valueOf(rs.getInt("br_num"));
            String br_street = rs.getString("br_street");
            String br_city = rs.getString("br_city");

            String tbData[] = {br_code, br_num, br_street, br_city};

            tbModel.addRow(tbData);
        }

        /*Phones*/
        stmt=con.createStatement();
        rs=stmt.executeQuery("select * from phones");
        tbModel = (DefaultTableModel) jTable1.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String ph_br_code =
String.valueOf(rs.getInt("ph_br_code"));
            String ph_number = rs.getString("ph_number");

            String tbData[] = {ph_br_code, ph_number};

            tbModel.addRow(tbData);
        }

        /*Destination*/
        stmt=con.createStatement();
        rs=stmt.executeQuery("select * from destination");
        tbModel = (DefaultTableModel) jTable3.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String dst_id = String.valueOf(rs.getInt("dst_id"));
            String dst_name = rs.getString("dst_name");
            String dst_descr = rs.getString("dst_descr");
            String dst_rtype = rs.getString("dst_rtype");
            String dst_language = rs.getString("dst_language");
            String dst_location =
String.valueOf(rs.getInt("dst_location"));
            String tbData[] = {dst_id, dst_name, dst_descr,
dst_rtype, dst_language, dst_location};

            tbModel.addRow(tbData);
        }

        /*Worker*/

```

```

stmt=con.createStatement();
rs=stmt.executeQuery("select * from worker");
tbModel = (DefaultTableModel)jTable4.getModel();
tbModel.setRowCount(0);
while(rs.next()){

    String wrk_AT = rs.getString("wrk_AT");
    String wrk_name = rs.getString("wrk_name");
    String wrk_lame = rs.getString("wrk_lame");
    String wrk_salary =
String.valueOf(rs.getFloat("wrk_salary") );
    String wrk_br_code = String.valueOf(
rs.getInt("wrk_br_code") );

    String tbData[] = {wrk_AT,wrk_name, wrk_lame, wrk_salary,
wrk_br_code};

    tbModel.addRow(tbData);
}

/*Guide*/
stmt=con.createStatement();
rs=stmt.executeQuery("select * from guide");
tbModel = (DefaultTableModel)jTable5.getModel();
tbModel.setRowCount(0);
while(rs.next()){

    String gui_AT = rs.getString("gui_AT");
    String gui_cv = rs.getString("gui_CV");

    String tbData[] = {gui_AT, gui_cv};

    tbModel.addRow(tbData);
}

/*Languages*/
stmt=con.createStatement();
rs=stmt.executeQuery("select * from languages");
tbModel = (DefaultTableModel)jTable6.getModel();
tbModel.setRowCount(0);
while(rs.next()){

    String lng_gui_AT = rs.getString("lng_gui_AT");
    String lng_language = rs.getString("lng_language");

    String tbData[] = {lng_gui_AT, lng_language};

    tbModel.addRow(tbData);
}

/*Driver*/
stmt=con.createStatement();
rs=stmt.executeQuery("select * from driver");
tbModel = (DefaultTableModel)jTable7.getModel();
tbModel.setRowCount(0);
while(rs.next()){

```

```

        String drv_AT = rs.getString("drv_AT");
        String drv_license = rs.getString("drv_license");
        String drv_route = rs.getString("drv_route");
        String drv_experience =
String.valueOf(rs.getInt("drv_experience") );

        String tbData[] = {drv_AT ,drv_license , drv_route,
drv_experience};

        tbModel.addRow(tbData);
    }

    /*Admin*/
    stmt=con.createStatement();
    rs=stmt.executeQuery("select * from admin");
    tbModel = (DefaultTableModel)jTable8.getModel();
    tbModel.setRowCount(0);
    while(rs.next()){

        String adm_AT = rs.getString("adm_AT");
        String adm_type = rs.getString("adm_type");
        String adm_diploma = rs.getString("adm_diploma");

        String tbData[] = {adm_AT, adm_type, adm_diploma};

        tbModel.addRow(tbData);
    }

    /*Manages*/
    stmt=con.createStatement();
    rs=stmt.executeQuery("select * from manages");
    tbModel = (DefaultTableModel)jTable9.getModel();
    tbModel.setRowCount(0);
    while(rs.next()){

        String mng_adm_AT = rs.getString("mng_adm_AT");
        String mng_br_code =
String.valueOf(rs.getInt("mng_br_code") );

        String tbData[] = {mng_adm_AT , mng_br_code};

        tbModel.addRow(tbData);
    }

    /*Trip*/
    stmt=con.createStatement();
    rs=stmt.executeQuery("select * from trip");
    tbModel = (DefaultTableModel)jTable10.getModel();
    tbModel.setRowCount(0);
    while(rs.next()){

        String tr_id = String.valueOf(rs.getInt("tr_id") );
        String tr_departure =
String.valueOf(rs.getDate("tr_departure") );
        String tr_return = String.valueOf(
rs.getDate("tr_return") );

```

```

        //String tr_departure = rs.getString("tr_departure");
        //String tr_return = rs.getString("tr_return");
        String tr_maxseats =
String.valueOf(rs.getInt("tr_maxseats") );

        String tr_cost = String.valueOf(rs.getFloat("tr_cost") );

        String tr_br_code =
String.valueOf(rs.getInt("tr_br_code") );
        String tr_gui_AT = rs.getString("tr_gui_AT");
        String tr_drv_AT = rs.getString("tr_drv_AT");

        String tbData[] = {tr_id , tr_departure, tr_return,
tr_maxseats, tr_cost, tr_br_code, tr_gui_AT,tr_drv_AT};

        tbModel.addRow(tbData);
    }

    /*Event*/
    stmt=con.createStatement();
    rs=stmt.executeQuery("select * from event");
    tbModel = (DefaultTableModel)jTable11.getModel();
    tbModel.setRowCount(0);
    while(rs.next()){

        String ev_tr_id = String.valueOf(rs.getInt("ev_tr_id") );
        String ev_start = rs.getString("ev_start");
        String ev_end = rs.getString("ev_end");
        String ev_descr = rs.getString("ev_descr");

        String tbData[] = {ev_tr_id, ev_start, ev_end, ev_descr};

        tbModel.addRow(tbData);
    }

    /*Reservation*/
    stmt=con.createStatement();
    rs=stmt.executeQuery("select * from reservation");
    tbModel = (DefaultTableModel)jTable12.getModel();
    tbModel.setRowCount(0);
    while(rs.next()){

        String res_tr_id = String.valueOf(rs.getInt("res_tr_id")
);
        String res_seatnum = String.valueOf(
rs.getInt("res_seatnum") );
        String res_name = rs.getString("res_name");
        String res_lname = rs.getString("res_lname");
        String res_isadult = rs.getString("res_isadult");

        String tbData[] = {res_tr_id, res_seatnum, res_name,
res_lname,res_isadult};

        tbModel.addRow(tbData);
    }

```

```

/*Offers*/
stmt=con.createStatement();
rs=stmt.executeQuery("select * from offers");
tbModel = (DefaultTableModel)jTable13.getModel();
tbModel.setRowCount(0);
while(rs.next()){

    String offer_id = String.valueOf( rs.getInt("offer_id")
);

    String offer_startDate = rs.getString("offer_startDate");
    String offer_endDate = rs.getString("offer_endDate");
    String cost_per_person =
String.valueOf(rs.getFloat("cost_per_person") );
    String offer_dst_id = String.valueOf(
rs.getInt("offer_dst_id") );

    String tbData[] = {offer_id, offer_startDate,
offer_endDate, cost_per_person ,offer_dst_id};

    tbModel.addRow(tbData);
}

/*Reservation Offers*/
stmt=con.createStatement();
rs=stmt.executeQuery("select * from reservation_offers");
tbModel = (DefaultTableModel)jTable14.getModel();
tbModel.setRowCount(0);
while(rs.next()){

    String res_offer_id =
String.valueOf(rs.getInt("res_offer_id") );
    String res_name = rs.getString("res_name");
    String res_lname = rs.getString("res_lname");
    String offer_id_trip = String.valueOf(
rs.getInt("offer_id_trip") );
    String res_deposit = String.valueOf(
rs.getFloat("res_deposit") );

    String tbData[] = {res_offer_id, res_name, res_lname,
offer_id_trip, res_deposit};

    tbModel.addRow(tbData);
}

/*itOfficer*/
stmt=con.createStatement();
rs=stmt.executeQuery("select * from itOfficer");
tbModel = (DefaultTableModel)jTable15.getModel();
tbModel.setRowCount(0);
while(rs.next()){

    String it_AT = rs.getString("it_AT");
    String password = rs.getString("password");
    String start_date = rs.getString("start_date");
    String end_date = rs.getString("end_date");

    String tbData[] = {it_AT, password, start_date,

```

```

end_date};

        tbModel.addRow(tbData);
    }

    /*Log*/
    stmt=con.createStatement();
    rs=stmt.executeQuery("select * from log");
    tbModel = (DefaultTableModel) jTable16.getModel();
    tbModel.setRowCount(0);
    while(rs.next()){

        String logid = String.valueOf(rs.getInt("logid"));
        String logdescrc = rs.getString("logdescrc");
        String log_ID_AT = rs.getString("log_ID_AT");
        String log_date = rs.getString("log_date");

        String tbData[] = {logid, logdescrc, log_ID_AT,
log_date};

        tbModel.addRow(tbData);
    }
    con.close();
}

catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);
}

}

/**
 * This method is called from within the constructor to initialize the
form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-
BEGIN: initComponents
    private void initComponents() {

        jSeparator4 = new javax.swing.JSeparator();
        dataMenuPanel = new javax.swing.JPanel();
        branchesPanel = new javax.swing.JPanel();
        jScrollPane2 = new javax.swing.JScrollPane();
        jTable2 = new javax.swing.JTable();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jLabel8 = new javax.swing.JLabel();
        jLabel9 = new javax.swing.JLabel();
        jLabel10 = new javax.swing.JLabel();
        jLabel11 = new javax.swing.JLabel();
        jTextField2 = new javax.swing.JTextField();
        jTextField3 = new javax.swing.JTextField();
        jTextField4 = new javax.swing.JTextField();
        jTextField5 = new javax.swing.JTextField();
        jButton5 = new javax.swing.JButton();

```

```
adminsPanel = new javax.swing.JPanel();
jScrollPane8 = new javax.swing.JScrollPane();
jTable8 = new javax.swing.JTable();
jButton18 = new javax.swing.JButton();
jButton19 = new javax.swing.JButton();
jButton20 = new javax.swing.JButton();
jLabel29 = new javax.swing.JLabel();
jLabel30 = new javax.swing.JLabel();
jLabel31 = new javax.swing.JLabel();
jComboBox4 = new javax.swing.JComboBox<>();
jComboBox5 = new javax.swing.JComboBox<>();
jTextField21 = new javax.swing.JTextField();
destinationsPanel = new javax.swing.JPanel();
jScrollPane3 = new javax.swing.JScrollPane();
jTable3 = new javax.swing.JTable();
jButton9 = new javax.swing.JButton();
jButton10 = new javax.swing.JButton();
jButton11 = new javax.swing.JButton();
jLabel17 = new javax.swing.JLabel();
jLabel18 = new javax.swing.JLabel();
jLabel19 = new javax.swing.JLabel();
jTextField10 = new javax.swing.JTextField();
jTextField11 = new javax.swing.JTextField();
jTextField12 = new javax.swing.JTextField();
jLabel20 = new javax.swing.JLabel();
jLabel21 = new javax.swing.JLabel();
jLabel22 = new javax.swing.JLabel();
jTextField14 = new javax.swing.JTextField();
jTextField15 = new javax.swing.JTextField();
jComboBox20 = new javax.swing.JComboBox<>();
driversPanel = new javax.swing.JPanel();
jScrollPane7 = new javax.swing.JScrollPane();
jTable7 = new javax.swing.JTable();
jButton21 = new javax.swing.JButton();
jButton22 = new javax.swing.JButton();
jButton23 = new javax.swing.JButton();
jLabel32 = new javax.swing.JLabel();
jLabel33 = new javax.swing.JLabel();
jLabel34 = new javax.swing.JLabel();
jLabel35 = new javax.swing.JLabel();
jComboBox6 = new javax.swing.JComboBox<>();
jComboBox21 = new javax.swing.JComboBox<>();
jTextField13 = new javax.swing.JTextField();
jComboBox22 = new javax.swing.JComboBox<>();
itOfficersPanel = new javax.swing.JPanel();
jScrollPane15 = new javax.swing.JScrollPane();
jTable15 = new javax.swing.JTable();
jButton43 = new javax.swing.JButton();
jButton44 = new javax.swing.JButton();
jLabel60 = new javax.swing.JLabel();
jComboBox17 = new javax.swing.JComboBox<>();
jLabel61 = new javax.swing.JLabel();
jTextField38 = new javax.swing.JTextField();
jLabel62 = new javax.swing.JLabel();
jTextField39 = new javax.swing.JTextField();
jLabel63 = new javax.swing.JLabel();
jTextField40 = new javax.swing.JTextField();
```



```
jSeparator1 = new javax.swing.JSeparator();
jButton3 = new javax.swing.JButton();
jLabel15 = new javax.swing.JLabel();
jLabel16 = new javax.swing.JLabel();
jLabel125 = new javax.swing.JLabel();
jLabel126 = new javax.swing.JLabel();
jLabel131 = new javax.swing.JLabel();
jLabel132 = new javax.swing.JLabel();
jLabel133 = new javax.swing.JLabel();
jLabel134 = new javax.swing.JLabel();
jTextField34 = new javax.swing.JTextField();
jTextField45 = new javax.swing.JTextField();
jTextField46 = new javax.swing.JTextField();
jTextField47 = new javax.swing.JTextField();
jTextField49 = new javax.swing.JTextField();
jTextField50 = new javax.swing.JTextField();
jTextField51 = new javax.swing.JTextField();
jComboBox29 = new javax.swing.JComboBox<>();
eventsPanel = new javax.swing.JPanel();
jScrollPane11 = new javax.swing.JScrollPane();
jTable11 = new javax.swing.JTable();
jButton12 = new javax.swing.JButton();
jButton13 = new javax.swing.JButton();
jButton14 = new javax.swing.JButton();
jLabel23 = new javax.swing.JLabel();
jLabel24 = new javax.swing.JLabel();
jLabel25 = new javax.swing.JLabel();
jLabel26 = new javax.swing.JLabel();
jTextField17 = new javax.swing.JTextField();
jTextField18 = new javax.swing.JTextField();
jTextField19 = new javax.swing.JTextField();
jComboBox15 = new javax.swing.JComboBox<>();
guidesPanel = new javax.swing.JPanel();
jScrollPane5 = new javax.swing.JScrollPane();
jTable5 = new javax.swing.JTable();
jButton24 = new javax.swing.JButton();
jButton25 = new javax.swing.JButton();
jButton26 = new javax.swing.JButton();
jLabel36 = new javax.swing.JLabel();
jComboBox7 = new javax.swing.JComboBox<>();
jLabel37 = new javax.swing.JLabel();
jTextField24 = new javax.swing.JTextField();
languagesPanel = new javax.swing.JPanel();
jScrollPane6 = new javax.swing.JScrollPane();
jTable6 = new javax.swing.JTable();
jButton30 = new javax.swing.JButton();
jButton31 = new javax.swing.JButton();
jButton32 = new javax.swing.JButton();
jLabel40 = new javax.swing.JLabel();
jLabel41 = new javax.swing.JLabel();
jTextField25 = new javax.swing.JTextField();
jComboBox10 = new javax.swing.JComboBox<>();
managersPanel = new javax.swing.JPanel();
jScrollPane9 = new javax.swing.JScrollPane();
jTable9 = new javax.swing.JTable();
jButton27 = new javax.swing.JButton();
jButton29 = new javax.swing.JButton();
```

```
jLabel138 = new javax.swing.JLabel();
jLabel139 = new javax.swing.JLabel();
jComboBox8 = new javax.swing.JComboBox<>();
jComboBox9 = new javax.swing.JComboBox<>();
jButton28 = new javax.swing.JButton();
offersPanel = new javax.swing.JPanel();
jScrollPane13 = new javax.swing.JScrollPane();
jTable13 = new javax.swing.JTable();
jButton39 = new javax.swing.JButton();
jButton40 = new javax.swing.JButton();
jButton41 = new javax.swing.JButton();
jLabel155 = new javax.swing.JLabel();
jLabel156 = new javax.swing.JLabel();
jLabel157 = new javax.swing.JLabel();
jLabel158 = new javax.swing.JLabel();
jLabel159 = new javax.swing.JLabel();
jComboBox16 = new javax.swing.JComboBox<>();
jTextField16 = new javax.swing.JTextField();
jTextField35 = new javax.swing.JTextField();
jTextField36 = new javax.swing.JTextField();
jTextField37 = new javax.swing.JTextField();
phonesPanel = new javax.swing.JPanel();
jScrollPane1 = new javax.swing.JScrollPane();
jTable1 = new javax.swing.JTable();
jButton15 = new javax.swing.JButton();
jButton16 = new javax.swing.JButton();
jButton17 = new javax.swing.JButton();
jLabel27 = new javax.swing.JLabel();
jComboBox3 = new javax.swing.JComboBox<>();
jLabel28 = new javax.swing.JLabel();
jTextField20 = new javax.swing.JTextField();
reservationsPanel = new javax.swing.JPanel();
jScrollPane12 = new javax.swing.JScrollPane();
jTable12 = new javax.swing.JTable();
jButton35 = new javax.swing.JButton();
jButton37 = new javax.swing.JButton();
jButton38 = new javax.swing.JButton();
jLabel150 = new javax.swing.JLabel();
jComboBox14 = new javax.swing.JComboBox<>();
jLabel151 = new javax.swing.JLabel();
jLabel152 = new javax.swing.JLabel();
jLabel153 = new javax.swing.JLabel();
jLabel154 = new javax.swing.JLabel();
jTextField31 = new javax.swing.JTextField();
jTextField32 = new javax.swing.JTextField();
jTextField33 = new javax.swing.JTextField();
jComboBox19 = new javax.swing.JComboBox<>();
reservationOffersPanel = new javax.swing.JPanel();
jScrollPane14 = new javax.swing.JScrollPane();
jTable14 = new javax.swing.JTable();
jButton45 = new javax.swing.JButton();
jButton46 = new javax.swing.JButton();
jButton47 = new javax.swing.JButton();
jLabel164 = new javax.swing.JLabel();
jTextField41 = new javax.swing.JTextField();
jLabel165 = new javax.swing.JLabel();
jTextField42 = new javax.swing.JTextField();
```

```
jLabel166 = new javax.swing.JLabel();
jTextField43 = new javax.swing.JTextField();
jLabel167 = new javax.swing.JLabel();
jComboBox18 = new javax.swing.JComboBox<>();
jLabel168 = new javax.swing.JLabel();
jTextField44 = new javax.swing.JTextField();
tripsPanel = new javax.swing.JPanel();
jScrollPane10 = new javax.swing.JScrollPane();
jTable10 = new javax.swing.JTable();
jButton33 = new javax.swing.JButton();
jButton34 = new javax.swing.JButton();
jButton36 = new javax.swing.JButton();
jLabel42 = new javax.swing.JLabel();
jLabel43 = new javax.swing.JLabel();
jLabel44 = new javax.swing.JLabel();
jLabel45 = new javax.swing.JLabel();
jLabel46 = new javax.swing.JLabel();
jLabel47 = new javax.swing.JLabel();
jLabel48 = new javax.swing.JLabel();
jLabel49 = new javax.swing.JLabel();
jComboBox11 = new javax.swing.JComboBox<>();
jComboBox12 = new javax.swing.JComboBox<>();
jComboBox13 = new javax.swing.JComboBox<>();
jTextField26 = new javax.swing.JTextField();
jTextField27 = new javax.swing.JTextField();
jTextField28 = new javax.swing.JTextField();
jTextField29 = new javax.swing.JTextField();
jTextField30 = new javax.swing.JTextField();
workersPanel = new javax.swing.JPanel();
jScrollPane4 = new javax.swing.JScrollPane();
jTable4 = new javax.swing.JTable();
jButton6 = new javax.swing.JButton();
jButton7 = new javax.swing.JButton();
jButton8 = new javax.swing.JButton();
jLabel112 = new javax.swing.JLabel();
jLabel113 = new javax.swing.JLabel();
jLabel114 = new javax.swing.JLabel();
jLabel115 = new javax.swing.JLabel();
jLabel116 = new javax.swing.JLabel();
jTextField6 = new javax.swing.JTextField();
jTextField7 = new javax.swing.JTextField();
jTextField8 = new javax.swing.JTextField();
jTextField9 = new javax.swing.JTextField();
jComboBox2 = new javax.swing.JComboBox<>();
travelToPanel = new javax.swing.JPanel();
jScrollPane23 = new javax.swing.JScrollPane();
jTable23 = new javax.swing.JTable();
jButton48 = new javax.swing.JButton();
jButton49 = new javax.swing.JButton();
jButton50 = new javax.swing.JButton();
jLabel169 = new javax.swing.JLabel();
jComboBox23 = new javax.swing.JComboBox<>();
jLabel170 = new javax.swing.JLabel();
jComboBox24 = new javax.swing.JComboBox<>();
jLabel171 = new javax.swing.JLabel();
jTextField22 = new javax.swing.JTextField();
jLabel172 = new javax.swing.JLabel();
```

```
jTextField23 = new javax.swing.JTextField();
branchesMenuPanel = new javax.swing.JPanel();
branchInfoPanel = new javax.swing.JPanel();
jScrollPane17 = new javax.swing.JScrollPane();
jTable17 = new javax.swing.JTable();
jScrollPane18 = new javax.swing.JScrollPane();
jTable18 = new javax.swing.JTable();
jLabel11 = new javax.swing.JLabel();
jComboBox1 = new javax.swing.JComboBox<>();
jLabel12 = new javax.swing.JLabel();
jLabel13 = new javax.swing.JLabel();
jScrollPane19 = new javax.swing.JScrollPane();
jTable19 = new javax.swing.JTable();
jLabel14 = new javax.swing.JLabel();
jScrollPane20 = new javax.swing.JScrollPane();
jTable20 = new javax.swing.JTable();
jSeparator6 = new javax.swing.JSeparator();
availableTripsPanel = new javax.swing.JPanel();
jLabel121 = new javax.swing.JLabel();
jComboBox28 = new javax.swing.JComboBox<>();
jLabel122 = new javax.swing.JLabel();
startDateTextField1 = new javax.swing.JTextField();
jLabel123 = new javax.swing.JLabel();
endDateTextField1 = new javax.swing.JTextField();
jButton51 = new javax.swing.JButton();
jLabel124 = new javax.swing.JLabel();
jScrollPane29 = new javax.swing.JScrollPane();
jTable27 = new javax.swing.JTable();
jSeparator5 = new javax.swing.JSeparator();
offersMenuPanel = new javax.swing.JPanel();
jButton4 = new javax.swing.JButton();
jLabel17 = new javax.swing.JLabel();
jTextField1 = new javax.swing.JTextField();
jScrollPane21 = new javax.swing.JScrollPane();
jTable21 = new javax.swing.JTable();
jScrollPane22 = new javax.swing.JScrollPane();
jTable22 = new javax.swing.JTable();
jSeparator2 = new javax.swing.JSeparator();
logsMenuPanel = new javax.swing.JPanel();
jScrollPane16 = new javax.swing.JScrollPane();
jTable16 = new javax.swing.JTable();
jLabel87 = new javax.swing.JLabel();
jSeparator3 = new javax.swing.JSeparator();
adminMenuPanel = new javax.swing.JPanel();
jLabel174 = new javax.swing.JLabel();
jComboBox25 = new javax.swing.JComboBox<>();
jLabel173 = new javax.swing.JLabel();
jLabel175 = new javax.swing.JLabel();
jLabel176 = new javax.swing.JLabel();
jLabel177 = new javax.swing.JLabel();
jLabel178 = new javax.swing.JLabel();
jLabel179 = new javax.swing.JLabel();
jScrollPane24 = new javax.swing.JScrollPane();
jTextArea1 = new javax.swing.JTextArea();
jLabel180 = new javax.swing.JLabel();
jLabel181 = new javax.swing.JLabel();
jLabel182 = new javax.swing.JLabel();
```

```
jLabel183 = new javax.swing.JLabel();
jLabel184 = new javax.swing.JLabel();
jLabel185 = new javax.swing.JLabel();
jLabel186 = new javax.swing.JLabel();
jSeparator7 = new javax.swing.JSeparator();
driverMenuPanel = new javax.swing.JPanel();
jLabel188 = new javax.swing.JLabel();
jComboBox26 = new javax.swing.JComboBox<>();
jLabel190 = new javax.swing.JLabel();
jLabel191 = new javax.swing.JLabel();
jLabel192 = new javax.swing.JLabel();
jLabel193 = new javax.swing.JLabel();
jLabel194 = new javax.swing.JLabel();
jLabel196 = new javax.swing.JLabel();
jLabel197 = new javax.swing.JLabel();
jLabel198 = new javax.swing.JLabel();
jLabel199 = new javax.swing.JLabel();
jLabel1100 = new javax.swing.JLabel();
jLabel1101 = new javax.swing.JLabel();
jLabel1102 = new javax.swing.JLabel();
jLabel1103 = new javax.swing.JLabel();
jLabel1104 = new javax.swing.JLabel();
jLabel1105 = new javax.swing.JLabel();
jLabel1106 = new javax.swing.JLabel();
jLabel195 = new javax.swing.JLabel();
jScrollPane25 = new javax.swing.JScrollPane();
jTable24 = new javax.swing.JTable();
jSeparator8 = new javax.swing.JSeparator();
GuideMenuPanel = new javax.swing.JPanel();
jLabel1107 = new javax.swing.JLabel();
jComboBox27 = new javax.swing.JComboBox<>();
jLabel1109 = new javax.swing.JLabel();
jLabel1110 = new javax.swing.JLabel();
jLabel1111 = new javax.swing.JLabel();
jLabel1112 = new javax.swing.JLabel();
jLabel1113 = new javax.swing.JLabel();
jLabel1114 = new javax.swing.JLabel();
jLabel1115 = new javax.swing.JLabel();
jLabel1116 = new javax.swing.JLabel();
jLabel1117 = new javax.swing.JLabel();
jLabel1118 = new javax.swing.JLabel();
jScrollPane26 = new javax.swing.JScrollPane();
jTable25 = new javax.swing.JTable();
jScrollPane27 = new javax.swing.JScrollPane();
jTable26 = new javax.swing.JTable();
jLabel1119 = new javax.swing.JLabel();
jScrollPane28 = new javax.swing.JScrollPane();
jTextArea2 = new javax.swing.JTextArea();
jLabel1120 = new javax.swing.JLabel();
jSeparator9 = new javax.swing.JSeparator();
jMenuBar1 = new javax.swing.JMenuBar();
jMenu1 = new javax.swing.JMenu();
jMenuItem2 = new javax.swing.JMenuItem();
jMenuItem1 = new javax.swing.JMenuItem();
jMenuItem3 = new javax.swing.JMenuItem();
jMenuItem4 = new javax.swing.JMenuItem();
jMenuItem5 = new javax.swing.JMenuItem();
```

```

jMenuItem6 = new javax.swing.JMenuItem();
jMenuItem7 = new javax.swing.JMenuItem();
jMenuItem8 = new javax.swing.JMenuItem();
jMenuItem9 = new javax.swing.JMenuItem();
jMenuItem10 = new javax.swing.JMenuItem();
jMenuItem11 = new javax.swing.JMenuItem();
jMenuItem12 = new javax.swing.JMenuItem();
jMenuItem13 = new javax.swing.JMenuItem();
jMenuItem14 = new javax.swing.JMenuItem();
jMenuItem15 = new javax.swing.JMenuItem();
jMenuItem19 = new javax.swing.JMenuItem();
jMenu2 = new javax.swing.JMenu();
jMenuItem17 = new javax.swing.JMenuItem();
jMenuItem23 = new javax.swing.JMenuItem();
jMenu3 = new javax.swing.JMenu();
jMenuItem16 = new javax.swing.JMenuItem();
jMenu4 = new javax.swing.JMenu();
jMenuItem18 = new javax.swing.JMenuItem();
jMenu5 = new javax.swing.JMenu();
jMenuItem20 = new javax.swing.JMenuItem();
jMenuItem21 = new javax.swing.JMenuItem();
jMenuItem22 = new javax.swing.JMenuItem();
jMenu6 = new javax.swing.JMenu();

setDefaultCloseOperation(javax.swing.WindowConstants.DO_NOTHING_ON_CLOSE);
setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
setPreferredSize(new java.awt.Dimension(1250, 850));
setSize(new java.awt.Dimension(1250, 850));
addWindowListener(new java.awt.event.WindowAdapter() {
    public void windowClosing(java.awt.event.WindowEvent evt) {
        formWindowClosing(evt);
    }
});
getContentPane().setLayout(new java.awt.CardLayout());

dataMenuPanel.setPreferredSize(new java.awt.Dimension(1149, 780));
dataMenuPanel.setLayout(new java.awt.CardLayout());

jTable2.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null},
        {null, null, null, null}
    },
    new String [] {
        "br_code", "br_num", "br_street", "br_city"
    }
) {
    Class[] types = new Class [] {
        java.lang.Integer.class, java.lang.Integer.class,
java.lang.String.class, java.lang.String.class
    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
});

```

```

jScrollPane2.setViewportViewView(jTable2);

jButton1.setText("New Branch");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

jButton2.setText("Delete Branch");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

jLabel8.setText("br_code");
jLabel9.setText("br_num");
jLabel10.setText("br_street");
jLabel11.setText("br_city");

jTextField2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jTextField2ActionPerformed(evt);
    }
});

jTextField4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jTextField4ActionPerformed(evt);
    }
});

jButton5.setText("Change Branch Data");
jButton5.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton5ActionPerformed(evt);
    }
});

javax.swing.GroupLayout branchesPanelLayout = new
javax.swing.GroupLayout(branchesPanel);
branchesPanel.setLayout(branchesPanelLayout);
branchesPanelLayout.setHorizontalGroup(

branchesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(branchesPanelLayout.createSequentialGroup()

.addGroup(branchesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(branchesPanelLayout.createSequentialGroup()
        .addGroup(branchesPanelLayout.createSequentialGroup()
            .addGap(90, 90, 90)

```

```

.addGroup(branchesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING)
                .addComponent(jButton2)
                .addComponent(jButton1,
javax.swing.GroupLayout.PREFERRED_SIZE, 138,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jButton5))
                .addGap(185, 185, 185)

.addGroup(branchesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.TRAILING, false)

.addGroup(branchesPanelLayout.createSequentialGroup()
                .addComponent(jLabel19)
                .addGap(18, 18, 18)
                .addComponent(jTextField3,
javax.swing.GroupLayout.PREFERRED_SIZE, 172,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addGroup(javax.swing.GroupLayout.Alignment.LEADING,
branchesPanelLayout.createSequentialGroup()
                .addComponent(jLabel18)
                .addGap(18, 18, 18)
                .addComponent(jTextField2))

.addGroup(javax.swing.GroupLayout.Alignment.LEADING,
branchesPanelLayout.createSequentialGroup()

.addGroup(branchesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING)
                .addComponent(jLabel10)
                .addComponent(jLabel11))
                .addGap(18, 18, 18)

.addGroup(branchesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING)
                .addComponent(jTextField4)
                .addComponent(jTextField5))))
                .addGroup(branchesPanelLayout.createSequentialGroup()
                .addGap(89, 89, 89)
                .addComponent(jScrollPane2,
javax.swing.GroupLayout.PREFERRED_SIZE, 1077,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addContainerGap(91, Short.MAX_VALUE)
        );
        branchesPanelLayout.setVerticalGroup(

branchesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
                .addGroup(branchesPanelLayout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jScrollPane2,
javax.swing.GroupLayout.PREFERRED_SIZE, 450,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(54, 54, 54)

.addGroup(branchesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali

```



```

ignment.BASELINE)
        .addComponent(jButton1)
        .addComponent(jLabel8)
        .addComponent(jTextField2,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

    .addGroup(branchesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
ignment.LEADING)
        .addGroup(branchesPanelLayout.createSequentialGroup()
            .addGap(34, 34, 34)

    .addGroup(branchesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
ignment.BASELINE)
        .addComponent(jLabel9)
        .addComponent(jTextField3,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(50, 50, 50))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
branchesPanelLayout.createSequentialGroup()

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jButton5)
        .addGap(40, 40, 40)))

    .addGroup(branchesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
ignment.BASELINE)
        .addComponent(jLabel10)
        .addComponent(jTextField4,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jButton2))
        .addGap(46, 46, 46)

    .addGroup(branchesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
ignment.BASELINE)
        .addComponent(jLabel11)
        .addComponent(jTextField5,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(388, Short.MAX_VALUE))
    );

    dataMenuPanel.add(branchesPanel, "card2");

    jTable8.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {

        },
        new String [] {
            "adm_AT", "adm_type", "adm_diploma"
        }
    ) {
        Class[] types = new Class [] {
            java.lang.String.class, java.lang.String.class,
java.lang.String.class

```

```

    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
});
jScrollPane8.setViewportViewView(jTable8);

jButton18.setText("New Admin");
jButton18.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton18ActionPerformed(evt);
    }
});

jButton19.setText("Change Admin Data");
jButton19.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton19ActionPerformed(evt);
    }
});

jButton20.setText("Delete Admin");
jButton20.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton20ActionPerformed(evt);
    }
});

jLabel29.setText("adm_AT");

jLabel30.setText("adm_type");

jLabel31.setText("adm_diploma");

jComboBox4.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));
jComboBox4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jComboBox4ActionPerformed(evt);
    }
});

jComboBox5.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jComboBox5ActionPerformed(evt);
    }
});

javax.swing.GroupLayout adminsPanelLayout = new
javax.swing.GroupLayout(adminsPanel);
adminsPanel.setLayout(adminsPanelLayout);
adminsPanelLayout.setHorizontalGroup(
adminsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(adminsPanelLayout.createSequentialGroup()

```

```

        .addContainerGap()
        .addComponent(jScrollPane8,
javax.swing.GroupLayout.DEFAULT_SIZE, 1245, Short.MAX_VALUE)
        .addContainerGap()
        .addGroup(adminsPanelLayout.createSequentialGroup()
        .addGap(37, 37, 37)

.addGroup(adminsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.TRAILING, false)
        .addComponent(jButton19,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 162, Short.MAX_VALUE)
        .addComponent(jButton18,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
        .addComponent(jButton20,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        .addGap(120, 120, 120)

.addGroup(adminsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)

.addGroup(adminsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.TRAILING, false)
        .addComponent(jLabel29,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
        .addComponent(jLabel30,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        .addComponent(jLabel31))
        .addGap(41, 41, 41)

.addGroup(adminsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING, false)
        .addComponent(jComboBox4,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox5, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jTextField21))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
    adminsPanelLayout.setVerticalGroup(

adminsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
        .addGroup(adminsPanelLayout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jScrollPane8,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addGap(55, 55, 55)

        .addGroup(adminsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
            .addComponent(jButton18)
            .addComponent(jLabel29)
            .addComponent(jComboBox4,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(39, 39, 39)

        .addGroup(adminsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
            .addComponent(jButton19)

        .addGroup(adminsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
            .addComponent(jLabel30)
            .addComponent(jComboBox5,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(40, 40, 40)

        .addGroup(adminsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
            .addComponent(jButton20)
            .addComponent(jLabel31)
            .addComponent(jTextField21,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(482, Short.MAX_VALUE)
    );

    dataMenuPanel.add(adminsPanel, "card3");

    jTable3.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null, null, null, null, null},
            {null, null, null, null, null, null},
            {null, null, null, null, null, null},
            {null, null, null, null, null, null}
        },
        new String [] {
            "dst_id", "dst_name", "dst_descr", "dst_rtype",
"dst_language", "dst_location"
        }
    ) {
        Class[] types = new Class [] {
            java.lang.Integer.class, java.lang.String.class,
java.lang.String.class, java.lang.String.class, java.lang.String.class,
java.lang.Integer.class
        };

        public Class getColumnClass(int columnIndex) {
            return types [columnIndex];
        }
    });

```

```

jScrollPane3.setViewportViewView(jTable3);

jButton9.setText("New destination");
jButton9.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton9ActionPerformed(evt);
    }
});

jButton10.setText("Change destination data");
jButton10.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton10ActionPerformed(evt);
    }
});

jButton11.setText("Delete destination");
jButton11.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton11ActionPerformed(evt);
    }
});

jLabel17.setText("dst_id");

jLabel18.setText("sdt_name");

jLabel19.setText("dst_descr");

jLabel20.setText("dst_rtype");

jLabel21.setText("dst_language");

jLabel22.setText("dst_location");

javax.swing.GroupLayout destinationsPanelLayout = new
javax.swing.GroupLayout(destinationsPanel);
destinationsPanel.setLayout(destinationsPanelLayout);
destinationsPanelLayout.setHorizontalGroup(

destinationsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.LEADING)
    .addGroup(destinationsPanelLayout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jScrollPane3,
javax.swing.GroupLayout.DEFAULT_SIZE, 1245, Short.MAX_VALUE)
        .addContainerGap()
        .addGroup(destinationsPanelLayout.createSequentialGroup()
            .addGap(66, 66, 66)

.addGroup(destinationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.LEADING, false)
            .addComponent(jButton11,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
            .addComponent(jButton10,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,

```

```

Short.MAX_VALUE)
        .addComponent(jButton9,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        .addGap(198, 198, 198)

.addGroup(destinationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.LEADING)
        .addComponent(jLabel17)
        .addComponent(jLabel18)
        .addComponent(jLabel19))
        .addGap(41, 41, 41)

.addGroup(destinationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.LEADING, false)
        .addComponent(jTextField10)
        .addComponent(jTextField11)
        .addComponent(jTextField12,
javax.swing.GroupLayout.DEFAULT_SIZE, 124, Short.MAX_VALUE))
        .addGap(151, 151, 151)

.addGroup(destinationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.LEADING)
        .addComponent(jLabel20)
        .addComponent(jLabel21)
        .addComponent(jLabel22))
        .addGap(39, 39, 39)

.addGroup(destinationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.LEADING)

.addGroup(destinationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.LEADING, false)
        .addComponent(jTextField14)
        .addComponent(jTextField15,
javax.swing.GroupLayout.DEFAULT_SIZE, 115, Short.MAX_VALUE))
        .addComponent(jComboBox20,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    );
    destinationsPanelLayout.setVerticalGroup(

destinationsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.LEADING)
        .addGroup(destinationsPanelLayout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jScrollPane3,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(73, 73, 73)

.addGroup(destinationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.BASELINE)
        .addComponent(jButton9)
        .addComponent(jLabel17)

```

```

        .addComponent(jTextField10,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel20)
        .addComponent(jComboBox20,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(46, 46, 46)

.addGroup(destinationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.LEADING)
        .addComponent(jButton10)

.addGroup(destinationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.BASELINE)
        .addComponent(jLabel18)
        .addComponent(jTextField11,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel21)
        .addComponent(jTextField14,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(45, 45, 45)

.addGroup(destinationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.BASELINE)
        .addComponent(jButton11)
        .addComponent(jLabel19)
        .addComponent(jTextField12,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel22)
        .addComponent(jTextField15,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(452, Short.MAX_VALUE))
    );

dataMenuPanel.add(destinationsPanel, "card4");

jTable7.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null}
    },
    new String [] {
        "drv_AT", "drv_license", "drv_route", "drv_experience"
    }
) {
    Class[] types = new Class [] {
        java.lang.String.class, java.lang.String.class,
java.lang.String.class, java.lang.Integer.class
    };

```

```

        public Class getColumnClass(int columnIndex) {
            return types [columnIndex];
        }
    });
    jScrollPane7.setViewportViewView(jTable7);

    jButton21.setText("New Driver");
    jButton21.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton21ActionPerformed(evt);
        }
    });

    jButton22.setText("Change Driver Data");
    jButton22.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton22ActionPerformed(evt);
        }
    });

    jButton23.setText("Delete Driver");
    jButton23.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton23ActionPerformed(evt);
        }
    });

    jLabel32.setText("drv_AT");

    jLabel33.setText("drv_license");

    jLabel34.setText("drv_route");

    jLabel35.setText("drv_experience");

    jComboBox6.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

    javax.swing.GroupLayout driversPanelLayout = new
javax.swing.GroupLayout(driversPanel);
    driversPanel.setLayout(driversPanelLayout);
    driversPanelLayout.setHorizontalGroup(

driversPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(driversPanelLayout.createSequentialGroup()
            .addGap(48, 48, 48)
            .addGroup(driversPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jButton21,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,

```



```

Short.MAX_VALUE)
        .addComponent(jButton22,
javax.swing.GroupLayout.DEFAULT_SIZE, 133, Short.MAX_VALUE)
        .addComponent(jButton23,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        .addGap(115, 115, 115)

.addGroup(driversPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
        .addComponent(jLabel32)
        .addComponent(jLabel33)
        .addComponent(jLabel34))
        .addGap(40, 40, 40)

.addGroup(driversPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
        .addComponent(jComboBox21,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(driversPanelLayout.createSequentialGroup())

.addGroup(driversPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.TRAILING, false)
        .addComponent(jComboBox22,
javax.swing.GroupLayout.Alignment.LEADING, 0, 171, Short.MAX_VALUE)
        .addComponent(jComboBox6,
javax.swing.GroupLayout.Alignment.LEADING, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addGap(107, 107, 107)
        .addComponent(jLabel35)
        .addGap(18, 18, 18)
        .addComponent(jTextField13,
javax.swing.GroupLayout.PREFERRED_SIZE, 233,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    );
    driversPanelLayout.setVerticalGroup(

driversPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
        .addGroup(driversPanelLayout.createSequentialGroup())
        .addContainerGap()
        .addComponent(jScrollPane7,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(47, 47, 47)

.addGroup(driversPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jButton21)
        .addComponent(jLabel32)
        .addComponent(jLabel35)
        .addComponent(jComboBox6,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addComponent(jTextField13,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(47, 47, 47)

.addGroup(driversPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.TRAILING)

.addGroup(driversPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)

        .addComponent(jButton22)
        .addComponent(jLabel133))
        .addGroup(driversPanelLayout.createSequentialGroup())
        .addComponent(jComboBox21,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(7, 7, 7)))

.addGroup(driversPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)

        .addGroup(driversPanelLayout.createSequentialGroup())
        .addGap(50, 50, 50)

.addGroup(driversPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)

        .addComponent(jButton23)
        .addComponent(jLabel134)))
        .addGroup(driversPanelLayout.createSequentialGroup())
        .addGap(39, 39, 39)
        .addComponent(jComboBox22,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(466, Short.MAX_VALUE)
    );

dataMenuPanel.add(driversPanel, "card5");

jTable15.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null}
    },
    new String [] {
        "it_AT", "password", "start_date", "end_date"
    }
));
jScrollPane15.setViewportViewView(jTable15);

jButton43.setText("Change it officer data");
jButton43.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton43ActionPerformed(evt);
    }
});

```

```

jButton44.setText("Delete it officer");
jButton44.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton44ActionPerformed(evt);
    }
});

jLabel60.setText("it_AT");

jComboBox17.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

jLabel61.setText("password");

jLabel62.setText("start_date");

jLabel63.setText("end_date");

jButton3.setText("Add It Officer");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});

jLabel5.setText("AT:");

jLabel6.setText("First Name:");

jLabel125.setText("Last Name:");

jLabel126.setText("Salary:");

jLabel131.setText("Branch Code:");

jLabel132.setText("Password:");

jLabel133.setText("Stard Date:");

jLabel134.setText("End Date:");

jTextField34.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jTextField34ActionPerformed(evt);
    }
});

jTextField45.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jTextField45ActionPerformed(evt);
    }
});

jTextField47.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jTextField47ActionPerformed(evt);
    }
});

```

```

    });

    jTextField50.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jTextField50ActionPerformed(evt);
        }
    });

    jComboBox29.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

    javax.swing.GroupLayout itOfficersPanelLayout = new
javax.swing.GroupLayout(itOfficersPanel);
    itOfficersPanel.setLayout(itOfficersPanelLayout);
    itOfficersPanelLayout.setHorizontalGroup(

itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
        .addGroup(itOfficersPanelLayout.createSequentialGroup()

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
            .addGroup(itOfficersPanelLayout.createSequentialGroup()
                .addComponent(jScrollPane15)
                .addGroup(itOfficersPanelLayout.createSequentialGroup()
                    .addGap(32, 32, 32)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                        .addComponent(jSeparator1,
javax.swing.GroupLayout.PREFERRED_SIZE, 1219,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGroup(itOfficersPanelLayout.createSequentialGroup()

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                            .addComponent(jButton44,
javax.swing.GroupLayout.PREFERRED_SIZE, 151,
javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(jButton43,
javax.swing.GroupLayout.PREFERRED_SIZE, 151,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)

.addGroup(itOfficersPanelLayout.createSequentialGroup()
                                    .addGap(147, 147, 147)
                                    .addComponent(jLabel61))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
itOfficersPanelLayout.createSequentialGroup()

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                    .addComponent(jLabel60))

```

```

        .addGap(18, 18, 18)

    .addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
        .addComponent(jComboBox17,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField38,
javax.swing.GroupLayout.PREFERRED_SIZE, 154,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(94, 94, 94)

    .addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.TRAILING)
        .addComponent(jLabel63)
        .addComponent(jLabel62))
        .addGap(18, 18, 18)

    .addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING, false)
        .addComponent(jTextField40,
javax.swing.GroupLayout.DEFAULT_SIZE, 187, Short.MAX_VALUE)
        .addComponent(jTextField39))

    .addGroup(itOfficersPanelLayout.createSequentialGroup())

    .addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
        .addComponent(jLabel5,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 29,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addGroup(itOfficersPanelLayout.createSequentialGroup())
        .addComponent(jButton3,
javax.swing.GroupLayout.PREFERRED_SIZE, 148,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(137, 137, 137)
        .addComponent(jLabel6))
        .addComponent(jLabel126,
javax.swing.GroupLayout.Alignment.TRAILING)
        .addComponent(jLabel125,
javax.swing.GroupLayout.Alignment.TRAILING))
        .addGap(18, 18, 18)

    .addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
        .addComponent(jTextField45,
javax.swing.GroupLayout.PREFERRED_SIZE, 187,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField47,
javax.swing.GroupLayout.PREFERRED_SIZE, 187,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField46,
javax.swing.GroupLayout.PREFERRED_SIZE, 187,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField34,

```

```

javax.swing.GroupLayout.PREFERRED_SIZE, 187,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(87, 87, 87)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.TRAILING)
        .addComponent(jLabel132)
        .addComponent(jLabel131)
        .addComponent(jLabel133)
        .addComponent(jLabel134))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
        .addComponent(jTextField49,
javax.swing.GroupLayout.PREFERRED_SIZE, 187,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField50,
javax.swing.GroupLayout.PREFERRED_SIZE, 187,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField51,
javax.swing.GroupLayout.PREFERRED_SIZE, 187,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGroup(itOfficersPanelLayout.createSequentialGroup()
        .addGap(21, 21, 21)
        .addComponent(jComboBox29,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))))))
        .addContainerGap()
    );
    itOfficersPanelLayout.setVerticalGroup(

itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
        .addGroup(itOfficersPanelLayout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jScrollPane15,
javax.swing.GroupLayout.PREFERRED_SIZE, 350,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(54, 54, 54)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
        .addComponent(jLabel160)
        .addComponent(jComboBox17,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel163)
        .addComponent(jTextField40,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jButton43))
        .addGap(18, 18, 18)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A

```

```

lignment.LEADING)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
    .addComponent(jButton44)
    .addComponent(jLabel61)
    .addComponent(jTextField38,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
    .addComponent(jTextField39,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jLabel62)))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
    .addComponent(jSeparator1,
javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(18, 18, 18)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
    .addComponent(jLabel5)
    .addComponent(jLabel131)
    .addComponent(jTextField34,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jComboBox29,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
    .addGroup(itOfficersPanelLayout.createSequentialGroup())

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
    .addComponent(jTextField45,
javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
    .addComponent(jLabel6)
    .addComponent(jTextField49,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jLabel132)))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A

```

```

lignment.LEADING)
        .addComponent(jLabel1133)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
        .addComponent(jLabel1125)
        .addComponent(jTextField46,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField50,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGroup(itOfficersPanelLayout.createSequentialGroup()
            .addGap(26, 26, 26)
            .addComponent(jButton3)))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(itOfficersPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
        .addComponent(jLabel1126)
        .addComponent(jLabel1134)
        .addComponent(jTextField47,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField51,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(477, Short.MAX_VALUE))
    );

dataMenuPanel.add(itOfficersPanel, "card6");

jTable11.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null}
    },
    new String [] {
        "ev_tr_id", "ev_start", "ev_end", "ev_descr"
    }
) {
    Class[] types = new Class [] {
        java.lang.Integer.class, java.lang.String.class,
java.lang.String.class, java.lang.String.class
    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
});
jScrollPane11.setViewportViewView(jTable11);

jButton12.setText("New Event");
jButton12.addActionListener(new java.awt.event.ActionListener() {

```



```

        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton12ActionPerformed(evt);
        }
    });

    jButton13.setText("Change event data");
    jButton13.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton13ActionPerformed(evt);
        }
    });

    jButton14.setText("Delete event data");
    jButton14.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton14ActionPerformed(evt);
        }
    });

    jLabel23.setText("ev_tr_id");

    jLabel24.setText("ev_start");

    jLabel25.setText("ev_end");

    jLabel26.setText("ev_descr");

    jComboBox15.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

    javax.swing.GroupLayout eventsPanelLayout = new
javax.swing.GroupLayout(eventsPanel);
    eventsPanel.setLayout(eventsPanelLayout);
    eventsPanelLayout.setHorizontalGroup(

eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(eventsPanelLayout.createSequentialGroup()

.addGroup(eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(eventsPanelLayout.createSequentialGroup()
                .addGroup(eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(eventsPanelLayout.createSequentialGroup()
                        .addContainerGap()
                        .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 1245, Short.MAX_VALUE))
                    .addGroup(eventsPanelLayout.createSequentialGroup()
                        .addGroup(eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
                            .addComponent(jButton14,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(jButton13,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(jButton12,

```

```

javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        .addGap(103, 103, 103)

.addGroup(eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
        .addComponent(jLabel123)
        .addComponent(jLabel124)
        .addComponent(jLabel125)
        .addComponent(jLabel126))
        .addGap(30, 30, 30)

.addGroup(eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)

.addGroup(eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING, false)
        .addComponent(jTextField17)
        .addComponent(jTextField18,
javax.swing.GroupLayout.DEFAULT_SIZE, 157, Short.MAX_VALUE))
        .addComponent(jTextField19,
javax.swing.GroupLayout.PREFERRED_SIZE, 199,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox15,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(0, 0, Short.MAX_VALUE)))
        .addContainerGap()
    );
    eventsPanelLayout.setVerticalGroup(

eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
        .addGroup(eventsPanelLayout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jScrollPane11,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(38, 38, 38)

.addGroup(eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jButton12)
        .addComponent(jLabel123)
        .addComponent(jComboBox15,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(43, 43, 43)

.addGroup(eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
        .addComponent(jButton13)

.addGroup(eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jLabel124)
        .addComponent(jTextField17,

```

```

javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(42, 42, 42)

    .addGroup(eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jButton14)
        .addComponent(jLabel25)
        .addComponent(jTextField18,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(46, 46, 46)

    .addGroup(eventsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
        .addComponent(jLabel26)
        .addComponent(jTextField19,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(425, Short.MAX_VALUE))
    );

    dataMenuPanel.add(eventsPanel, "card7");

    jTable5.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null},
            {null, null},
            {null, null},
            {null, null}
        },
        new String [] {
            "gui_AT", "gui_cv"
        }
    ) {
        Class[] types = new Class [] {
            java.lang.String.class, java.lang.String.class
        };

        public Class getColumnClass(int columnIndex) {
            return types [columnIndex];
        }
    });
    jScrollPane5.setViewportViewView(jTable5);

    jButton24.setText("New Guide");
    jButton24.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton24ActionPerformed(evt);
        }
    });

    jButton25.setText("Change Guide Data");
    jButton25.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton25ActionPerformed(evt);
        }
    });

```

```

    });

    jButton26.setText("Delete Guide");
    jButton26.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton26ActionPerformed(evt);
        }
    });

    jLabel36.setText("gui_At");

    JComboBox7.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));
    JComboBox7.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            JComboBox7ActionPerformed(evt);
        }
    });

    jLabel37.setText("gui_cv");

    javax.swing.GroupLayout guidesPanelLayout = new
javax.swing.GroupLayout(guidesPanel);
    guidesPanel.setLayout(guidesPanelLayout);
    guidesPanelLayout.setHorizontalGroup(

guidesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(guidesPanelLayout.createSequentialGroup()
            .addGap(49, 49, 49)
            .addComponent(jButton24,
javax.swing.GroupLayout.DEFAULT_SIZE, 1245, Short.MAX_VALUE)
            .addGap(49, 49, 49)
            .addGroup(guidesPanelLayout.createSequentialGroup()
                .addComponent(jButton25,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                .addComponent(jButton26,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                .addGap(157, 157, 157)
            )
        )
        .addGroup(guidesPanelLayout.createSequentialGroup()
            .addGap(27, 27, 27)
            .addComponent(jLabel36)
            .addComponent(jLabel37)
            .addGap(27, 27, 27)
        )
        .addGroup(guidesPanelLayout.createSequentialGroup()
            .addGap(27, 27, 27)
            .addComponent(jButton26)
            .addGap(27, 27, 27)
        )
    );
    guidesPanelLayout.setVerticalGroup(
        guidesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(guidesPanelLayout.createSequentialGroup()
                .addGap(49, 49, 49)
                .addComponent(jButton24)
                .addGap(49, 49, 49)
                .addGroup(guidesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(jButton25)
                    .addComponent(jButton26)
                )
                .addGap(157, 157, 157)
            )
            .addGroup(guidesPanelLayout.createSequentialGroup()
                .addGap(27, 27, 27)
                .addComponent(jLabel36)
                .addComponent(jLabel37)
                .addGap(27, 27, 27)
            )
            .addGroup(guidesPanelLayout.createSequentialGroup()
                .addGap(27, 27, 27)
                .addComponent(jButton26)
                .addGap(27, 27, 27)
            )
    );

```

```

        .addComponent(jTextField24,
javax.swing.GroupLayout.PREFERRED_SIZE, 287,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox7,
javax.swing.GroupLayout.PREFERRED_SIZE, 198,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
    guidesPanelLayout.setVerticalGroup(

guidesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(guidesPanelLayout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jScrollPane5,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(49, 49, 49)

        .addGroup(guidesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

            .addComponent(jButton24)
            .addComponent(jLabel36)
            .addComponent(jComboBox7,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(49, 49, 49)
            .addComponent(jButton25)
            .addGap(14, 14, 14)

        .addGroup(guidesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

            .addComponent(jLabel37)
            .addComponent(jTextField24,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(11, 11, 11)
            .addComponent(jButton26)
            .addContainerGap(471, Short.MAX_VALUE))
    );

    dataMenuPanel.add(guidesPanel, "card8");

    jTable6.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null},
            {null, null},
            {null, null},
            {null, null}
        },
        new String [] {
            "lng_gui_AT", "lng_language"
        }
    ) {
        Class[] types = new Class [] {
            java.lang.String.class, java.lang.String.class

```

```

        };

        public Class getColumnClass(int columnIndex) {
            return types [columnIndex];
        }
    });
    jScrollPane6.setViewportViewView(jTable6);

    jButton30.setText("New language");
    jButton30.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton30ActionPerformed(evt);
        }
    });

    jButton31.setText("Change language data");
    jButton31.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton31ActionPerformed(evt);
        }
    });

    jButton32.setText("Delete language");
    jButton32.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton32ActionPerformed(evt);
        }
    });

    jLabel40.setText("lng_gui_AT");

    jLabel41.setText("lng_language");

    jComboBox10.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

    javax.swing.GroupLayout languagesPanelLayout = new
javax.swing.GroupLayout(languagesPanel);
    languagesPanel.setLayout(languagesPanelLayout);
    languagesPanelLayout.setHorizontalGroup(

languagesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(languagesPanelLayout.createSequentialGroup()
            .addGap(10, 10, 10)
            .addComponent(jScrollPane6,
javax.swing.GroupLayout.DEFAULT_SIZE, 1245, Short.MAX_VALUE)
            .addGap(10, 10, 10)
            .addGroup(languagesPanelLayout.createSequentialGroup()
                .addGap(73, 73, 73)

            .addGroup(languagesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
                .addComponent(jButton30,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                .addComponent(jButton31,

```

```

javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
        .addComponent(jButton32,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        .addGap(167, 167, 167)

    .addGroup(languagesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Al
ignment.LEADING)
        .addComponent(jLabel41)
        .addComponent(jLabel40))
        .addGap(73, 73, 73)

    .addGroup(languagesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Al
ignment.LEADING, false)
        .addComponent(jTextField25)
        .addComponent(jComboBox10, 0, 225, Short.MAX_VALUE))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
    languagesPanelLayout.setVerticalGroup(

languagesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LE
ADING)
        .addGroup(languagesPanelLayout.createSequentialGroup())
        .addContainerGap()
        .addComponent(jScrollPane6,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(68, 68, 68)

    .addGroup(languagesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Al
ignment.BASELINE)
        .addComponent(jButton30)
        .addComponent(jLabel40)
        .addComponent(jComboBox10,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(69, 69, 69)

    .addGroup(languagesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Al
ignment.BASELINE)
        .addComponent(jButton31)
        .addComponent(jLabel41)
        .addComponent(jTextField25,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(68, 68, 68)
        .addComponent(jButton32)
        .addContainerGap(411, Short.MAX_VALUE))
    );

    dataMenuPanel.add(languagesPanel, "card9");

    jTable9.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null},

```

```

        {null, null},
        {null, null},
        {null, null}
    },
    new String [] {
        "mng_adm_AT", "mng_br_code"
    }
) {
    Class[] types = new Class [] {
        java.lang.String.class, java.lang.Integer.class
    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
});
jScrollPane9.setViewportViewView(jTable9);

jButton27.setText("New Manager");
jButton27.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton27ActionPerformed(evt);
    }
});

jButton29.setText("Delete Manager");
jButton29.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton29ActionPerformed(evt);
    }
});

jLabel38.setText("mng_adm_AT");
jLabel39.setText("mng_br_code");

jComboBox8.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

jComboBox9.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

jButton28.setText("Change Manager");
jButton28.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton28ActionPerformed(evt);
    }
});

javax.swing.GroupLayout managersPanelLayout = new
javax.swing.GroupLayout(managersPanel);
managersPanel.setLayout(managersPanelLayout);
managersPanelLayout.setHorizontalGroup(
managersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(managersPanelLayout.createSequentialGroup()

```



```

        .addContainerGap()
        .addComponent(jScrollPane9,
javax.swing.GroupLayout.DEFAULT_SIZE, 1245, Short.MAX_VALUE)
        .addContainerGap()
        .addGroup(managersPanelLayout.createSequentialGroup())
        .addGap(59, 59, 59)

    .addGroup(managersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING)
        .addComponent(jButton28,
javax.swing.GroupLayout.PREFERRED_SIZE, 146,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(managersPanelLayout.createSequentialGroup())

    .addGroup(managersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING)
        .addComponent(jButton29)

    .addGroup(managersPanelLayout.createSequentialGroup())
        .addComponent(jButton27)
        .addGap(255, 255, 255)

    .addGroup(managersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING)
        .addComponent(jLabel39)
        .addComponent(jLabel38)))
        .addGap(36, 36, 36)

    .addGroup(managersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING, false)
        .addComponent(jComboBox8, 0, 161,
Short.MAX_VALUE)
        .addComponent(jComboBox9, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    );
    managersPanelLayout.setVerticalGroup(
managersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING)
        .addGroup(managersPanelLayout.createSequentialGroup())
        .addContainerGap()
        .addComponent(jScrollPane9,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(64, 64, 64)

    .addGroup(managersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.BASELINE)
        .addComponent(jButton27)
        .addComponent(jLabel38)
        .addComponent(jComboBox8,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(54, 54, 54)
        .addComponent(jButton28)

```

```

        .addGap(12, 12, 12)

.addGroup(managersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.BASELINE)
        .addComponent(jLabel139)
        .addComponent(jComboBox9,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(21, 21, 21)
        .addComponent(jButton29)
        .addContainerGap(443, Short.MAX_VALUE))
);

dataMenuPanel.add(managersPanel, "card10");

jTable13.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null}
    },
    new String [] {
        "offer_id", "offer_startDate", "offer_endDate",
"cost_per_person", "offer_dst_id"
    }
));
jTable13.addAncestorListener(new javax.swing.event.AncestorListener()
{
    public void ancestorAdded(javax.swing.event.AncestorEvent evt) {
        jTable13AncestorAdded(evt);
    }
    public void ancestorMoved(javax.swing.event.AncestorEvent evt) {
    }
    public void ancestorRemoved(javax.swing.event.AncestorEvent evt)
{
    }
});
jScrollPane13.setViewportViewView(jTable13);

jButton39.setText("New Offer");
jButton39.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton39ActionPerformed(evt);
    }
});

jButton40.setText("Change Offer Data");
jButton40.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton40ActionPerformed(evt);
    }
});

jButton41.setText("Delete Offer");
jButton41.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {

```

```

        jButton41ActionPerformed(evt);
    }
});

jLabel155.setText("offer_id");

jLabel156.setText("offer_startDate");

jLabel157.setText("offer_endDate");

jLabel158.setText("cost_per_person");

jLabel159.setText("offer_dst_id");

jComboBox16.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

javax.swing.GroupLayout offersPanelLayout = new
javax.swing.GroupLayout(offersPanel);
offersPanel.setLayout(offersPanelLayout);
offersPanelLayout.setHorizontalGroup(

offersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(offersPanelLayout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jScrollPane13,
javax.swing.GroupLayout.DEFAULT_SIZE, 1245, Short.MAX_VALUE)
        .addContainerGap()
        .addGroup(offersPanelLayout.createSequentialGroup()
            .addGap(50, 50, 50)

.addGroup(offersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
            .addComponent(jButton40)
            .addComponent(jButton39,
javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jButton41,
javax.swing.GroupLayout.Alignment.LEADING))
            .addGap(138, 138, 138)

.addGroup(offersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(offersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
            .addComponent(jLabel156,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
            .addComponent(jLabel155,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
            .addComponent(jLabel157))
            .addGap(33, 33, 33)

.addGroup(offersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

```

```

        .addComponent(jTextField16)
        .addComponent(jTextField35)
        .addComponent(jTextField36,
javax.swing.GroupLayout.DEFAULT_SIZE, 136, Short.MAX_VALUE))
        .addGap(84, 84, 84)

.addGroup(offersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING, false)
        .addComponent(jLabel159,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
        .addComponent(jLabel158,
javax.swing.GroupLayout.DEFAULT_SIZE, 93, Short.MAX_VALUE))
        .addGap(36, 36, 36)

.addGroup(offersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
        .addComponent(jComboBox16,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField37,
javax.swing.GroupLayout.PREFERRED_SIZE, 201,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    );
    offersPanelLayout.setVerticalGroup(

offersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
        .addGroup(offersPanelLayout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jScrollPane13,
javax.swing.GroupLayout.PREFERRED_SIZE, 348,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(41, 41, 41)

.addGroup(offersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jButton39)
        .addComponent(jLabel155)
        .addComponent(jLabel158)
        .addComponent(jTextField16,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField37,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(67, 67, 67)

.addGroup(offersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jButton40)
        .addComponent(jLabel156)
        .addComponent(jLabel159)
        .addComponent(jComboBox16,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,

```

```

javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField35,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(60, 60, 60)

.addGroup(offersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
        .addComponent(jButton41)

.addGroup(offersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jLabel157)
        .addComponent(jTextField36,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addContainerGap(527, Short.MAX_VALUE))
    );

dataMenuPanel.add(offersPanel, "card11");

jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null},
        {null, null},
        {null, null},
        {null, null}
    },
    new String [] {
        "ph_br_code", "ph_number"
    }
) {
    Class[] types = new Class [] {
        java.lang.Integer.class, java.lang.String.class
    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
});
jScrollPane1.setViewportViewView(jTable1);

jButton15.setText("New Phone Number");
jButton15.addComponentListener(new java.awt.event.ComponentAdapter()
{
    public void componentResized(java.awt.event.ComponentEvent evt) {
        jButton15ComponentResized(evt);
    }
});
jButton15.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton15ActionPerformed(evt);
    }
});

jButton16.setText("Change Phone Data");
jButton16.addActionListener(new java.awt.event.ActionListener() {

```

```

        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton16ActionPerformed(evt);
        }
    });

    jButton17.setText("Delete Phone ");
    jButton17.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton17ActionPerformed(evt);
        }
    });

    jLabel27.setText("ph_br_code");

    jComboBox3.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

    jLabel28.setText("ph_number");

    javax.swing.GroupLayout phonesPanelLayout = new
javax.swing.GroupLayout(phonesPanel);
    phonesPanel.setLayout(phonesPanelLayout);
    phonesPanelLayout.setHorizontalGroup(

phonesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(phonesPanelLayout.createSequentialGroup()
            .addGap(40, 40, 40)
            .addGroup(phonesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jButton15,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                .addComponent(jButton16,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                .addComponent(jButton17,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
            .addGap(145, 145, 145)
            .addGroup(phonesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addComponent(jLabel27,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                .addComponent(jLabel28,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
            .addGap(50, 50, 50)
        )
    );

```

```

.addGroup(phonesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
    .addComponent(jComboBox3,
javax.swing.GroupLayout.PREFERRED_SIZE, 93,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jTextField20,
javax.swing.GroupLayout.PREFERRED_SIZE, 128,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
);
phonesPanelLayout.setVerticalGroup(

phonesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
    .addGroup(phonesPanelLayout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jScrollPane1,
javax.swing.GroupLayout.PREFERRED_SIZE, 408,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(49, 49, 49)

    .addGroup(phonesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jButton15)
        .addComponent(jLabel27)
        .addComponent(jComboBox3,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(54, 54, 54)

    .addGroup(phonesPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jButton16)
        .addComponent(jLabel28)
        .addComponent(jTextField20,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(54, 54, 54)
        .addComponent(jButton17)
        .addContainerGap(478, Short.MAX_VALUE))
);

dataMenuPanel.add(phonesPanel, "card12");

jTable12.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null}
    },
    new String [] {
        "res_tr_id", "res_seatnum", "res_name", "res_lname",
"res_isadult"
    }
) {

```

```

        Class[] types = new Class [] {
            java.lang.Integer.class, java.lang.Object.class,
            java.lang.String.class, java.lang.String.class, java.lang.String.class
        };

        public Class getColumnClass(int columnIndex) {
            return types [columnIndex];
        }
    });
    jScrollPane12.setViewportViewView(jTable12);

    jButton35.setText("New reservation");
    jButton35.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton35ActionPerformed(evt);
        }
    });

    jButton37.setText("Change reservation data");
    jButton37.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton37ActionPerformed(evt);
        }
    });

    jButton38.setText("Delete reservation");
    jButton38.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton38ActionPerformed(evt);
        }
    });

    jLabel50.setText("res_tr_id");

    jComboBox14.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

    jLabel51.setText("res_seatnum");

    jLabel52.setText("res_name");

    jLabel53.setText("res_lname");

    jLabel54.setText("res_isadult");

    javax.swing.GroupLayout reservationsPanelLayout = new
javax.swing.GroupLayout(reservationsPanel);
    reservationsPanel.setLayout(reservationsPanelLayout);
    reservationsPanelLayout.setHorizontalGroup(

reservationsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.LEADING)
        .addGroup(reservationsPanelLayout.createSequentialGroup()

.addGroup(reservationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.LEADING)
        .addGroup(reservationsPanelLayout.createSequentialGroup()

```



```

        .addContainerGap()
        .addComponent(jScrollPane12,
javax.swing.GroupLayout.DEFAULT_SIZE, 1245, Short.MAX_VALUE))
        .addGroup(reservationsPanelLayout.createSequentialGroup()
            .addGap(64, 64, 64)

        .addGroup(reservationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
            .Alignment.LEADING)
            .addComponent(jButton35)
            .addComponent(jButton37)
            .addComponent(jButton38))
            .addGap(113, 113, 113)

        .addGroup(reservationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
            .Alignment.LEADING)
            .addComponent(jLabel150,
javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel151)
            .addComponent(jLabel152))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addGroup(reservationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
            .Alignment.LEADING, false)
            .addComponent(jComboBox14,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField31,
javax.swing.GroupLayout.DEFAULT_SIZE, 141, Short.MAX_VALUE)
            .addComponent(jTextField32))
            .addGap(79, 79, 79)

        .addGroup(reservationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
            .Alignment.LEADING)
            .addComponent(jLabel153)
            .addComponent(jLabel154))
            .addGap(33, 33, 33)

        .addGroup(reservationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
            .Alignment.LEADING)
            .addComponent(jTextField33,
javax.swing.GroupLayout.PREFERRED_SIZE, 162,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jComboBox19,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(0, 0, Short.MAX_VALUE))
            .addContainerGap()
    );
    reservationsPanelLayout.setVerticalGroup(
reservationsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.LEADING)
    .addGroup(reservationsPanelLayout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jScrollPane12,

```

```

javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(62, 62, 62)

.addGroup(reservationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.BASELINE)
        .addComponent(jButton35)
        .addComponent(jLabel150)
        .addComponent(jComboBox14,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel153)
        .addComponent(jTextField33,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(53, 53, 53)

.addGroup(reservationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.BASELINE)
        .addComponent(jButton37)
        .addComponent(jLabel151)
        .addComponent(jLabel154)
        .addComponent(jTextField31,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox19,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(63, 63, 63)

.addGroup(reservationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.TRAILING)
        .addComponent(jButton38)

.addGroup(reservationsPanelLayout.createParallelGroup(javax.swing.GroupLayout
.Alignment.BASELINE)
        .addComponent(jLabel152)
        .addComponent(jTextField32,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addContainerGap(438, Short.MAX_VALUE)
    );

dataMenuPanel.add(reservationsPanel, "card13");

jTable14.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null}
    },
    new String [] {
        "res_offer_id", "res_name", "res_lname", "offer_id_trip",
"res_deposit"
    }
));

```

```

jScrollPane14.setViewportViewView(jTable14);

jButton45.setText("New Reservation Offer");
jButton45.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton45ActionPerformed(evt);
    }
});

jButton46.setText("Change Reservation Offer Data");
jButton46.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton46ActionPerformed(evt);
    }
});

jButton47.setText("Delete Reservation Offer");
jButton47.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton47ActionPerformed(evt);
    }
});

jLabel64.setText("res_offer_id");

jLabel65.setText("res_name");

jLabel66.setText("res_lname");

jLabel67.setText("offer_id_trip");

jComboBox18.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

jLabel68.setText("res_deposit");

javax.swing.GroupLayout reservationOffersPanelLayout = new
javax.swing.GroupLayout(reservationOffersPanel);
reservationOffersPanel.setLayout(reservationOffersPanelLayout);
reservationOffersPanelLayout.setHorizontalGroup(

reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
    .addGroup(reservationOffersPanelLayout.createSequentialGroup()

.addGroup(reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupL
ayout.Alignment.LEADING)

.addGroup(reservationOffersPanelLayout.createSequentialGroup()
    .addComponent(jScrollPane14,
javax.swing.GroupLayout.DEFAULT_SIZE, 1245, Short.MAX_VALUE)

.addGroup(reservationOffersPanelLayout.createSequentialGroup()
    .addGap(42, 42, 42)

.addGroup(reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupL

```

```

ayout.Alignment.LEADING)

.addGroup(reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupLayout.
ayout.Alignment.LEADING, false)
        .addComponent(jButton45,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
        .addComponent(jButton46,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        .addComponent(jButton47,
javax.swing.GroupLayout.PREFERRED_SIZE, 195,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(146, 146, 146)

.addGroup(reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupLayout.
ayout.Alignment.LEADING)
        .addComponent(jLabel64)
        .addComponent(jLabel65)
        .addComponent(jLabel66))
        .addGap(18, 18, 18)

.addGroup(reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupLayout.
ayout.Alignment.LEADING, false)
        .addComponent(jTextField41)
        .addComponent(jTextField42)
        .addComponent(jTextField43,
javax.swing.GroupLayout.DEFAULT_SIZE, 159, Short.MAX_VALUE))
        .addGap(130, 130, 130)

.addGroup(reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupLayout.
ayout.Alignment.LEADING)
        .addComponent(jLabel67)
        .addComponent(jLabel68))
        .addGap(18, 18, 18)

.addGroup(reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupLayout.
ayout.Alignment.LEADING)
        .addComponent(jComboBox18,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField44,
javax.swing.GroupLayout.PREFERRED_SIZE, 124,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(0, 0, Short.MAX_VALUE)))
        .addContainerGap()
    );
    reservationOffersPanelLayout.setVerticalGroup(

reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.LEADING)
        .addGroup(reservationOffersPanelLayout.createSequentialGroup())
        .addContainerGap()
        .addComponent(jScrollPane14,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(44, 44, 44)

```

```

.addGroup(reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jButton45)
    .addComponent(jLabel164)
    .addComponent(jTextField41,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jLabel167)
    .addComponent(jComboBox18,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGap(42, 42, 42)

.addGroup(reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jButton46)

.addGroup(reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel165)
    .addComponent(jTextField42,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jLabel168)
    .addComponent(jTextField44,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))

.addGroup(reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(reservationOffersPanelLayout.createSequentialGroup())
    .addGap(51, 51, 51)

.addGroup(reservationOffersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jButton47)
    .addComponent(jLabel166)))

.addGroup(reservationOffersPanelLayout.createSequentialGroup())
    .addGap(42, 42, 42)
    .addComponent(jTextField43,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
    .addContainerGap(479, Short.MAX_VALUE))
);

dataMenuPanel.add(reservationOffersPanel, "card14");

jTable10.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null, null, null, null, null},
        {null, null, null, null, null, null, null, null},
        {null, null, null, null, null, null, null, null},
        {null, null, null, null, null, null, null, null}
    },

```

```

        new String [] {
            "tr_id", "tr_departure", "tr_return", "tr_maxseats",
            "tr_cost", "tr_br_code", "tr_gui_AT", "tr_drv_AT"
        }
    ) {
        Class[] types = new Class [] {
            java.lang.String.class, java.lang.String.class,
            java.lang.String.class, java.lang.String.class, java.lang.String.class,
            java.lang.String.class, java.lang.String.class, java.lang.String.class
        };

        public Class getColumnClass(int columnIndex) {
            return types [columnIndex];
        }
    });
jScrollPane10.setViewportView(jTable10);

jButton33.setText("Change trip data");
jButton33.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton33ActionPerformed(evt);
    }
});

jButton34.setText("Delete trip");
jButton34.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton34ActionPerformed(evt);
    }
});

jButton36.setText("New trip");
jButton36.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton36ActionPerformed(evt);
    }
});

jLabel42.setText("tr_id");

jLabel43.setText("tr_departure");

jLabel44.setText("tr_return");

jLabel45.setText("tr_cost");

jLabel46.setText("tr_br_code");

jLabel47.setText("tr_gui_AT");

jLabel48.setText("tr_maxseats");

jLabel49.setText("tr_drv_AT");

jComboBox11.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

```

```
jComboBox12.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

jComboBox13.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

jTextField26.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jTextField26ActionPerformed(evt);
    }
});

javax.swing.GroupLayout tripsPanelLayout = new
javax.swing.GroupLayout(tripsPanel);
tripsPanel.setLayout(tripsPanelLayout);
tripsPanelLayout.setHorizontalGroup(

tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(tripsPanelLayout.createSequentialGroup()

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(tripsPanelLayout.createSequentialGroup()
        .addGroup(tripsPanelLayout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jScrollPane10,
javax.swing.GroupLayout.DEFAULT_SIZE, 1245, Short.MAX_VALUE)
            .addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                .addGap(72, 72, 72)

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
    .addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addGroup(tripsPanelLayout.createSequentialGroup()
        .addComponent(jButton36,
javax.swing.GroupLayout.PREFERRED_SIZE, 140,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(106, 106, 106)
        .addComponent(jLabel42))

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
    .addGroup(tripsPanelLayout.createSequentialGroup()
        .addComponent(jButton33,
javax.swing.GroupLayout.PREFERRED_SIZE, 140,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jButton34,
javax.swing.GroupLayout.PREFERRED_SIZE, 140,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
```

```

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
    .addComponent(jLabel44,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    .addComponent(jLabel43,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)))
    .addComponent(jLabel48,
javax.swing.GroupLayout.PREFERRED_SIZE, 79,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(35, 35, 35)

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
    .addComponent(jTextField28,
javax.swing.GroupLayout.DEFAULT_SIZE, 136, Short.MAX_VALUE)
    .addComponent(jTextField27)
    .addComponent(jTextField26)
    .addComponent(jTextField29,
javax.swing.GroupLayout.PREFERRED_SIZE, 124,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(70, 70, 70)

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
    .addComponent(jLabel49,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    .addComponent(jLabel45)
    .addComponent(jLabel46,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    .addComponent(jLabel47))

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(tripsPanelLayout.createSequentialGroup()
    .addGap(53, 53, 53)

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jComboBox11,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jComboBox12,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jComboBox13,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))

```



```

.addGroup(tripsPanelLayout.createSequentialGroup()
            .addGap(18, 18, 18)
            .addComponent(jTextField30,
javax.swing.GroupLayout.PREFERRED_SIZE, 168,
javax.swing.GroupLayout.PREFERRED_SIZE)))
            .addGap(0, 0, Short.MAX_VALUE)))
        .addContainerGap()
    );
    tripsPanelLayout.setVerticalGroup(

tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(tripsPanelLayout.createSequentialGroup()
                .addContainerGap()

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel47)
            .addComponent(jComboBox12,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(tripsPanelLayout.createSequentialGroup()
                .addComponent(jScrollPane10,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(85, 85, 85)

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jButton36)
            .addComponent(jLabel42)
            .addComponent(jLabel45)
            .addComponent(jTextField26,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField30,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(40, 40, 40)

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jButton33)
            .addComponent(jLabel43)
            .addComponent(jLabel46)
            .addComponent(jComboBox11,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jTextField27,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(45, 45, 45)

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment

```

```

ent.LEADING)
        .addComponent(jButton34)

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.BASELINE)
        .addComponent(jLabel44)
        .addComponent(jTextField28,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(53, 53, 53)

.addGroup(tripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignm
ent.BASELINE)
        .addComponent(jLabel48)
        .addComponent(jLabel49)
        .addComponent(jComboBox13,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField29,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(371, Short.MAX_VALUE)
    );

dataMenuPanel.add(tripsPanel, "card15");

workersPanel.addAncestorListener(new
javax.swing.event.AncestorListener() {
    public void ancestorAdded(javax.swing.event.AncestorEvent evt) {
        workersPanelAncestorAdded(evt);
    }
    public void ancestorMoved(javax.swing.event.AncestorEvent evt) {
    }
    public void ancestorRemoved(javax.swing.event.AncestorEvent evt)
{
    }
});

jTable4.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null}
    },
    new String [] {
        "wrk_AT", "wrk_name", "wrk_lame", "wrk_salary", "wrk_br-code"
    }
) {
    Class[] types = new Class [] {
        java.lang.String.class, java.lang.String.class,
java.lang.String.class, java.lang.String.class, java.lang.String.class
    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
}

```

```

});
jScrollPane4.setViewportViewView(jTable4);

jButton6.setText("New Worker");
jButton6.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton6ActionPerformed(evt);
    }
});

jButton7.setText("Change Worker Data");
jButton7.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton7ActionPerformed(evt);
    }
});

jButton8.setText("Delete Worker");
jButton8.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton8ActionPerformed(evt);
    }
});

jLabel12.setText("wrk_AT");

jLabel13.setText("wrk_name");

jLabel14.setText("wrk_lname");

jLabel15.setText("wrk_salary");

jLabel16.setText("wrk_br_code");

jTextField7.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jTextField7ActionPerformed(evt);
    }
});

jComboBox2.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

javax.swing.GroupLayout workersPanelLayout = new
javax.swing.GroupLayout(workersPanel);
workersPanel.setLayout(workersPanelLayout);
workersPanelLayout.setHorizontalGroup(

workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(workersPanelLayout.createSequentialGroup()
        .addGap(10, 10, 10)
        .addComponent(jScrollPane4,
javax.swing.GroupLayout.DEFAULT_SIZE, 1245, Short.MAX_VALUE)
        .addGap(10, 10, 10)
        .addGroup(workersPanelLayout.createSequentialGroup()
            .addGap(10, 10, 10)

```

```

.addGroup(workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
    .addComponent(jButton6,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    .addComponent(jButton7,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    .addComponent(jButton8,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    .addGap(335, 335, 335)

.addGroup(workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
    .addGroup(workersPanelLayout.createSequentialGroup()
        .addComponent(jLabel16)
        .addGap(48, 48, 48)
        .addComponent(jComboBox2,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGroup(workersPanelLayout.createSequentialGroup()
        .addComponent(jLabel15)
        .addGap(60, 60, 60)
        .addComponent(jTextField9,
javax.swing.GroupLayout.DEFAULT_SIZE, 169, Short.MAX_VALUE))
    .addGroup(workersPanelLayout.createSequentialGroup()
        .addComponent(jLabel14)
        .addGap(57, 57, 57)
        .addComponent(jTextField8))
    .addGroup(workersPanelLayout.createSequentialGroup()

.addGroup(workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jLabel13)
    .addComponent(jLabel12))
    .addGap(60, 60, 60)

.addGroup(workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
    .addComponent(jTextField6)
    .addComponent(jTextField7,
javax.swing.GroupLayout.DEFAULT_SIZE, 169, Short.MAX_VALUE)))
    .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    );
workersPanelLayout.setVerticalGroup(

workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(workersPanelLayout.createSequentialGroup()
        .addContainerGap()

.addGroup(workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
    .addGroup(workersPanelLayout.createSequentialGroup()

```

```

        .addComponent(jScrollPane4,
javax.swing.GroupLayout.PREFERRED_SIZE, 446,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(60, 60, 60)
        .addComponent(jButton6))

.addGroup(workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jLabel12)
        .addComponent(jTextField6,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))

.addGroup(workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
        .addGroup(workersPanelLayout.createSequentialGroup())
        .addGap(54, 54, 54)
        .addComponent(jButton7)
        .addGap(50, 50, 50))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
workersPanelLayout.createSequentialGroup())

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jLabel13)
        .addComponent(jTextField7,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(33, 33, 33)

.addGroup(workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jLabel14)
        .addComponent(jTextField8,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(17, 17, 17)))

.addGroup(workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
        .addComponent(jButton8)
        .addGroup(workersPanelLayout.createSequentialGroup())
        .addGap(18, 18, 18)

.addGroup(workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jLabel15)
        .addComponent(jTextField9,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGap(41, 41, 41)

.addGroup(workersPanelLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jLabel16)

```

```

        .addComponent(jComboBox2,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(353, Short.MAX_VALUE)
    );

    dataMenuPanel.add(workersPanel, "card16");

    jTable23.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null},
            {null, null, null, null}
        },
        new String [] {
            "to_tr_id", "to_dst_id", "to_arrival", "to_departure"
        }
    ) {
        Class[] types = new Class [] {
            java.lang.String.class, java.lang.String.class,
java.lang.String.class, java.lang.Object.class
        };

        public Class getColumnClass(int columnIndex) {
            return types [columnIndex];
        }
    });
    jScrollPane23.setViewportViewView(jTable23);

    jButton48.setText("New travelTo");
    jButton48.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton48ActionPerformed(evt);
        }
    });

    jButton49.setText("Change travelTo");
    jButton49.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton49ActionPerformed(evt);
        }
    });

    jButton50.setText("Delete travelTo");
    jButton50.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jButton50ActionPerformed(evt);
        }
    });

    jLabel69.setText("to_tr_id");

    jComboBox23.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

    jLabel70.setText("to_dst_id");

```

```

        jComboBox24.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

        jLabel71.setText("to_arrival");

        jTextField22.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jTextField22ActionPerformed(evt);
            }
        });

        jLabel72.setText("to_departure");

        javax.swing.GroupLayout travelToPanelLayout = new
javax.swing.GroupLayout(travelToPanel);
        travelToPanel.setLayout(travelToPanelLayout);
        travelToPanelLayout.setHorizontalGroup(

travelToPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(travelToPanelLayout.createSequentialGroup()

.addGroup(travelToPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING)
                .addGroup(travelToPanelLayout.createSequentialGroup()
                    .addContainerGap()
                    .addComponent(jScrollPane23,
javax.swing.GroupLayout.PREFERRED_SIZE, 1129,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGroup(travelToPanelLayout.createSequentialGroup()
                        .addGap(53, 53, 53)

.addGroup(travelToPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING, false)
                            .addComponent(jButton48,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                            .addComponent(jButton49,
javax.swing.GroupLayout.DEFAULT_SIZE, 169, Short.MAX_VALUE)
                            .addComponent(jButton50,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                            .addGap(173, 173, 173)

.addGroup(travelToPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING)
                                .addComponent(jLabel169,
javax.swing.GroupLayout.PREFERRED_SIZE, 105,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                .addComponent(jLabel170,
javax.swing.GroupLayout.PREFERRED_SIZE, 65,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                .addGap(18, 18, 18)

.addGroup(travelToPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING)

```

```

        .addComponent(jComboBox23,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox24,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(111, 111, 111)

.addGroup(travelToPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING, false)

.addGroup(travelToPanelLayout.createSequentialGroup()
        .addComponent(jLabel71,
javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addComponent(jTextField22,
javax.swing.GroupLayout.PREFERRED_SIZE, 166,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addGroup(travelToPanelLayout.createSequentialGroup()
        .addComponent(jLabel72)
        .addGap(18, 18, 18)
        .addComponent(jTextField23))))
        .addContainerGap(122, Short.MAX_VALUE)
);
travelToPanelLayout.setVerticalGroup(

travelToPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
        .addGroup(travelToPanelLayout.createSequentialGroup()
                .addGap(15, 15, 15)
                .addComponent(jScrollPane23,
javax.swing.GroupLayout.PREFERRED_SIZE, 298,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(43, 43, 43)

.addGroup(travelToPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.BASELINE)
        .addComponent(jButton48)
        .addComponent(jLabel69)
        .addComponent(jComboBox23,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel71)
        .addComponent(jTextField22,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(26, 26, 26)
        .addComponent(jButton49)
        .addGap(3, 3, 3)

.addGroup(travelToPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.BASELINE)
        .addComponent(jLabel70)
        .addComponent(jComboBox24,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,

```



```

javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel172)
        .addComponent(jTextField23,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(4, 4, 4)
        .addComponent(jButton50)
        .addContainerGap(638, Short.MAX_VALUE))
    );

    dataMenuPanel.add(travelToPanel, "card17");

    getContentPane().add(dataMenuPanel, "card2");

    branchesMenuPanel.setLayout(new java.awt.CardLayout());

    jTable17.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null}
        },
        new String [] {
            "First Name", "Last Name", "Salary"
        }
    ) {
        Class[] types = new Class [] {
            java.lang.String.class, java.lang.String.class,
java.lang.String.class
        };

        public Class getColumnClass(int columnIndex) {
            return types [columnIndex];
        }
    });
    jScrollPane17.setViewportViewView(jTable17);

    jTable18.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null, null, null, null, null, null, null},
            {null, null, null, null, null, null, null, null},
            {null, null, null, null, null, null, null, null},
            {null, null, null, null, null, null, null, null}
        },
        new String [] {
            "Branch Code", "Number", "Street", "City", "admin (FN)",
"Last name (LN)", "Num of reservations", "Income"
        }
    ) {
        Class[] types = new Class [] {
            java.lang.String.class, java.lang.String.class,
java.lang.String.class, java.lang.String.class, java.lang.String.class,
java.lang.String.class, java.lang.String.class, java.lang.String.class
        };

        public Class getColumnClass(int columnIndex) {

```

```

        return types [columnIndex];
    }
});
jScrollPane18.setViewportViewView(jTable18);

jLabel11.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel11.setText("Choose branch : ");

jComboBox1.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));
jComboBox1.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        jComboBox1MouseClicked(evt);
    }
});
jComboBox1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jComboBox1ActionPerformed(evt);
    }
});

jLabel2.setFont(new java.awt.Font("Cantarell", 1, 24)); // NOI18N
jLabel2.setText("Branch Info");

jLabel3.setFont(new java.awt.Font("Cantarell", 1, 24)); // NOI18N
jLabel3.setText("Worker's info");

jTable19.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null},
        {null},
        {null},
        {null}
    },
    new String [] {
        "Total salary of workers"
    }
) {
    Class[] types = new Class [] {
        java.lang.String.class
    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
});
jScrollPane19.setViewportViewView(jTable19);

jLabel4.setFont(new java.awt.Font("Cantarell", 1, 24)); // NOI18N
jLabel4.setText("Financial");

jTable20.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null},
        {null, null, null},
        {null, null, null},
        {null, null, null}
    }

```

```

        },
        new String [] {
            "Income", "Expenses", "Turnover"
        }
    ));
    jTable20.getTableHeader().setReorderingAllowed(false);
    jScrollPane20.setViewportViewView(jTable20);

    javax.swing.GroupLayout branchInfoPanelLayout = new
javax.swing.GroupLayout(branchInfoPanel);
    branchInfoPanel.setLayout(branchInfoPanelLayout);
    branchInfoPanelLayout.setHorizontalGroup(

branchInfoPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
        .addGroup(branchInfoPanelLayout.createSequentialGroup())
            .addGap(22, 22, 22)

        .addGroup(branchInfoPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
            .addComponent(jLabel3,
javax.swing.GroupLayout.PREFERRED_SIZE, 167,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel4)
            .addComponent(jScrollPane20,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGroup(branchInfoPanelLayout.createSequentialGroup())
                .addComponent(jScrollPane17,
javax.swing.GroupLayout.PREFERRED_SIZE, 897,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jScrollPane19,
javax.swing.GroupLayout.PREFERRED_SIZE, 174,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGroup(branchInfoPanelLayout.createSequentialGroup())
                .addComponent(jLabel11)
                .addGap(18, 18, 18)
                .addComponent(jComboBox1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel2,
javax.swing.GroupLayout.PREFERRED_SIZE, 167,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jSeparator6,
javax.swing.GroupLayout.PREFERRED_SIZE, 1206,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jScrollPane18,
javax.swing.GroupLayout.PREFERRED_SIZE, 1120,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(29, Short.MAX_VALUE))
        );
    branchInfoPanelLayout.setVerticalGroup(

branchInfoPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)

```

```

        .addGroup(branchInfoPanelLayout.createSequentialGroup())
        .addGap(27, 27, 27)

    .addGroup(branchInfoPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
        .addComponent(jLabel1)
        .addComponent(jComboBox1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jSeparator6,
javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jLabel2)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jScrollPane18,
javax.swing.GroupLayout.PREFERRED_SIZE, 92,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addGroup(branchInfoPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
branchInfoPanelLayout.createSequentialGroup()
            .addComponent(jLabel4)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jScrollPane20,
javax.swing.GroupLayout.PREFERRED_SIZE, 109,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jLabel3)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jScrollPane17,
javax.swing.GroupLayout.PREFERRED_SIZE, 328,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addComponent(jScrollPane19,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 83,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(371, 371, 371))
    );

    branchesMenuPanel.add(branchInfoPanel, "card3");

    jLabel121.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
    jLabel121.setText("Choose branch : ");

    jComboBox28.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));

```

```

jComboBox28.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        jComboBox28MouseClicked(evt);
    }
});

jComboBox28.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jComboBox28ActionPerformed(evt);
    }
});

jLabel122.setFont(new java.awt.Font("Cantarell", 1, 15)); // NOI18N
jLabel122.setText("Start Date");

startDateTextField1.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        startDateTextField1ActionPerformed(evt);
    }
});

jLabel123.setFont(new java.awt.Font("Cantarell", 1, 15)); // NOI18N
jLabel123.setText("End Date");

endDateTextField1.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        endDateTextField1ActionPerformed(evt);
    }
});

jButton51.setText("Show available trips");
jButton51.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton51ActionPerformed(evt);
    }
});

jLabel124.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel124.setText("Available Trips in given days");

jTable27.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null, null, null, null, null, null, null},
        {null, null, null, null, null, null, null, null, null, null},
        {null, null, null, null, null, null, null, null, null, null},
        {null, null, null, null, null, null, null, null, null, null}
    },
    new String [] {
        "tripCost", "maxSeats", "reservCount", "emptySeats",
        "drivName", "drivLname", "GuideLname", "GuideNam", "departureDate",
        "returnDate"
    }
) {
    Class[] types = new Class [] {
        java.lang.String.class, java.lang.String.class,
        java.lang.String.class, java.lang.String.class, java.lang.String.class,

```

```

java.lang.String.class, java.lang.String.class, java.lang.String.class,
java.lang.String.class, java.lang.String.class
    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
});
jScrollPane29.setViewportViewView(jTable27);

javax.swing.GroupLayout availableTripsPanelLayout = new
javax.swing.GroupLayout(availableTripsPanel);
availableTripsPanel.setLayout(availableTripsPanelLayout);
availableTripsPanelLayout.setHorizontalGroup(

availableTripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignme
nt.LEADING)
    .addGroup(availableTripsPanelLayout.createSequentialGroup()
        .addGap(36, 36, 36)

.addGroup(availableTripsPanelLayout.createParallelGroup(javax.swing.GroupLayo
ut.Alignment.LEADING)
    .addComponent(jScrollPane29,
javax.swing.GroupLayout.PREFERRED_SIZE, 1079,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGroup(availableTripsPanelLayout.createSequentialGroup()

.addGroup(availableTripsPanelLayout.createParallelGroup(javax.swing.GroupLayo
ut.Alignment.TRAILING)
    .addComponent(jLabel124,
javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(availableTripsPanelLayout.createSequentialGroup()
    .addComponent(jLabel121)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
    .addComponent(jComboBox28,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(18, 18, 18)
    .addComponent(jLabel122)))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(startDateTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, 142,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(24, 24, 24)
    .addComponent(jLabel123)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(endDateTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, 141,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(34, 34, 34)
    .addComponent(jButton51))
    .addComponent(jSeparator5,

```

```

javax.swing.GroupLayout.PREFERRED_SIZE, 1184,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(37, Short.MAX_VALUE))
    );
    availableTripsPanelLayout.setVerticalGroup(

availableTripsPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignme
nt.LEADING)
        .addGroup(availableTripsPanelLayout.createSequentialGroup())
        .addGap(23, 23, 23)

.addGroup(availableTripsPanelLayout.createParallelGroup(javax.swing.GroupLayo
ut.Alignment.TRAILING)

.addGroup(availableTripsPanelLayout.createParallelGroup(javax.swing.GroupLayo
ut.Alignment.BASELINE)
        .addComponent(jLabel122)
        .addComponent(startDateTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addGroup(availableTripsPanelLayout.createParallelGroup(javax.swing.GroupLayo
ut.Alignment.BASELINE)
        .addComponent(jLabel123)
        .addComponent(endDateTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jButton51))

.addGroup(availableTripsPanelLayout.createParallelGroup(javax.swing.GroupLayo
ut.Alignment.BASELINE)
        .addComponent(jLabel121)
        .addComponent(jComboBox28,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)
        .addComponent(jSeparator5,
javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jLabel124)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jScrollPane29,
javax.swing.GroupLayout.PREFERRED_SIZE, 239,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(761, Short.MAX_VALUE))
    );

    branchesMenuPanel.add(availableTripsPanel, "card3");

    getContentPane().add(branchesMenuPanel, "card3");

    jButton4.setText("Show reservations");
    jButton4.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {

```

```

        jButton4ActionPerformed(evt);
    }
});

jLabel7.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel7.setText("Customer Name:");

jTextField1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jTextField1ActionPerformed(evt);
    }
});

jTable21.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null, null, null, null},
        {null, null, null, null, null, null, null},
        {null, null, null, null, null, null, null},
        {null, null, null, null, null, null, null}
    },
    new String [] {
        "res_name", "res_lname", "offer_id", "offer_startDate",
"offer_endDate", "cost_per_person", "offer_dst_id"
    }
) {
    Class[] types = new Class [] {
        java.lang.String.class, java.lang.String.class,
java.lang.String.class, java.lang.String.class, java.lang.String.class,
java.lang.String.class, java.lang.String.class
    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
});
jScrollPane21.setViewportViewView(jTable21);

jTable22.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null, null, null},
        {null, null, null, null, null, null},
        {null, null, null, null, null, null},
        {null, null, null, null, null, null}
    },
    new String [] {
        "offer_id", "offer_startDate", "offer_endDate",
"cost_per_person", "offer_dst_id", "count(res_lname)"
    }
) {
    Class[] types = new Class [] {
        java.lang.String.class, java.lang.String.class,
java.lang.String.class, java.lang.String.class, java.lang.String.class,
java.lang.String.class
    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
});

```



```

    }
    });
    jScrollPane22.setViewportViewView(jTable22);

    javax.swing.GroupLayout offersMenuPanelLayout = new
javax.swing.GroupLayout(offersMenuPanel);
    offersMenuPanel.setLayout(offersMenuPanelLayout);
    offersMenuPanelLayout.setHorizontalGroup(

offersMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
        .addGroup(offersMenuPanelLayout.createSequentialGroup()
            .addGap(56, 56, 56)

.addGroup(offersMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING, false)
            .addGroup(offersMenuPanelLayout.createSequentialGroup()
                .addComponent(jLabel7)
                .addGap(18, 18, 18)
                .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, 170,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(32, 32, 32)
                .addComponent(jButton4,
javax.swing.GroupLayout.PREFERRED_SIZE, 153,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(separator2)
                .addComponent(jScrollPane21,
javax.swing.GroupLayout.DEFAULT_SIZE, 1136, Short.MAX_VALUE)
                .addComponent(jScrollPane22)
                .addContainerGap(65, Short.MAX_VALUE)
            );
    offersMenuPanelLayout.setVerticalGroup(

offersMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
        .addGroup(offersMenuPanelLayout.createSequentialGroup()
            .addGap(33, 33, 33)

.addGroup(offersMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
            .addComponent(jLabel7)
            .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jButton4)
            .addGap(19, 19, 19)
            .addComponent(separator2,
javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jScrollPane21,
javax.swing.GroupLayout.PREFERRED_SIZE, 289,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(48, 48, 48)
            .addComponent(jScrollPane22,
javax.swing.GroupLayout.PREFERRED_SIZE, 183,

```

```

javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(492, Short.MAX_VALUE))
    );

    getContentPane().add(offersMenuPanel, "card4");

    jTable16.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null}
        },
        new String [] {
            "Description", "IT's Last Name", "log_date"
        }
    ) {
        Class[] types = new Class [] {
            java.lang.String.class, java.lang.String.class,
java.lang.String.class
        };

        public Class getColumnClass(int columnIndex) {
            return types [columnIndex];
        }
    });
    jScrollPane16.setViewportViewView(jTable16);

    jLabel187.setFont(new java.awt.Font("Cantarell", 1, 36)); // NOI18N
    jLabel187.setText("LOGS");

    javax.swing.GroupLayout logsMenuPanelLayout = new
javax.swing.GroupLayout(logsMenuPanel);
    logsMenuPanel.setLayout(logsMenuPanelLayout);
    logsMenuPanelLayout.setHorizontalGroup(

logsMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
        .addGroup(logsMenuPanelLayout.createSequentialGroup()

.addGroup(logsMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
gnment.LEADING)
            .addGroup(logsMenuPanelLayout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jScrollPane16,
javax.swing.GroupLayout.DEFAULT_SIZE, 1245, Short.MAX_VALUE)
                .addGroup(logsMenuPanelLayout.createSequentialGroup()
                    .addGap(33, 33, 33)
                    .addComponent(jLabel187)
                    .addGap(0, 0, Short.MAX_VALUE)
                    .addGroup(logsMenuPanelLayout.createSequentialGroup()
                        .addContainerGap()
                        .addComponent(jSeparator3))
                    .addContainerGap()
                )
            )
        )
    );
    logsMenuPanelLayout.setVerticalGroup(

```

```

logsMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(logsMenuPanelLayout.createSequentialGroup()
        .addGap(33, 33, 33)
        .addComponent(jLabel187)
        .addGap(2, 2, 2)
        .addComponent(jSeparator3,
javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addComponent(jScrollPane16,
javax.swing.GroupLayout.PREFERRED_SIZE, 409,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(599, Short.MAX_VALUE)
    );

getContentPane().add(logsMenuPanel, "card5");

jLabel74.setFont(new java.awt.Font("Cantarell", 1, 36)); // NOI18N
jLabel74.setText("Admin");

jComboBox25.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));
jComboBox25.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jComboBox25ActionPerformed(evt);
    }
});

jLabel73.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel73.setText("First Name:");

jLabel75.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel75.setText("Last Name:");

jLabel76.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel76.setText("AT:");

jLabel77.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel77.setText("Branch Code:");

jLabel78.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel78.setText("Salary:");

jLabel79.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel79.setText("Type:");

jTextArea1.setColumns(20);
jTextArea1.setRows(5);
jScrollPane24.setViewportView(jTextArea1);

jLabel80.setFont(new java.awt.Font("Cantarell", 1, 24)); // NOI18N
jLabel80.setText("Diploma:");

jLabel81.setText("adm_name");

jLabel82.setText("adm_lname");

```

```

        jLabel83.setText("adm_at");

        jLabel84.setText("adm_br");

        jLabel85.setText("adm_salary");

        jLabel86.setText("adm_type");

        javax.swing.GroupLayout adminMenuPanelLayout = new
javax.swing.GroupLayout(adminMenuPanel);
        adminMenuPanel.setLayout(adminMenuPanelLayout);
        adminMenuPanelLayout.setHorizontalGroup(

adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addGroup(adminMenuPanelLayout.createSequentialGroup())
                    .addGap(64, 64, 64)

                .addGroup(adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                    .addGroup(adminMenuPanelLayout.createSequentialGroup())

                .addGroup(adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                    .addComponent(jLabel177,
javax.swing.GroupLayout.Alignment.TRAILING)
                    .addComponent(jLabel175,
javax.swing.GroupLayout.Alignment.TRAILING)
                    .addComponent(jLabel173,
javax.swing.GroupLayout.Alignment.TRAILING)
                    .addComponent(jLabel176,
javax.swing.GroupLayout.Alignment.TRAILING)
                    .addComponent(jLabel178,
javax.swing.GroupLayout.Alignment.TRAILING)
                    .addComponent(jLabel179,
javax.swing.GroupLayout.Alignment.TRAILING))
                    .addGap(24, 24, 24)

                .addGroup(adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

                    .addComponent(jLabel181,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)

                    .addComponent(jLabel182,
javax.swing.GroupLayout.DEFAULT_SIZE, 115, Short.MAX_VALUE)
                    .addComponent(jLabel183,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)

                    .addComponent(jLabel184,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)

                    .addComponent(jLabel185,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)

                    .addComponent(jLabel186,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,

```

```

Short.MAX_VALUE))
        .addGap(131, 131, 131)

.addGroup(adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jScrollPane24,
javax.swing.GroupLayout.PREFERRED_SIZE, 603,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel180)))
        .addGroup(adminMenuPanelLayout.createSequentialGroup())
        .addComponent(jLabel174)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jComboBox25,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addComponent(jSeparator7,
javax.swing.GroupLayout.PREFERRED_SIZE, 1120,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(73, Short.MAX_VALUE))
    );
    adminMenuPanelLayout.setVerticalGroup(

adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(adminMenuPanelLayout.createSequentialGroup())
        .addGap(29, 29, 29)

.addGroup(adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
        .addComponent(jLabel174,
javax.swing.GroupLayout.PREFERRED_SIZE, 29,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jComboBox25,
javax.swing.GroupLayout.PREFERRED_SIZE, 44,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)
        .addComponent(jSeparator7,
javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel173)
        .addComponent(jLabel180)
        .addComponent(jLabel181))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(adminMenuPanelLayout.createSequentialGroup())

.addGroup(adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

```

```

                .addComponent(jLabel182)
                .addComponent(jLabel175,
javax.swing.GroupLayout.PREFERRED_SIZE, 22,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
ignment.BASELINE)
                .addComponent(jLabel176)
                .addComponent(jLabel183))
        .addGap(9, 9, 9)

.addGroup(adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
ignment.BASELINE)
                .addComponent(jLabel184)
                .addComponent(jLabel177))
        .addGap(9, 9, 9)

.addGroup(adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
ignment.BASELINE)
                .addComponent(jLabel178)
                .addComponent(jLabel185))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(adminMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Ali
ignment.BASELINE)
                .addComponent(jLabel179)
                .addComponent(jLabel186))
        .addComponent(jScrollPane24,
javax.swing.GroupLayout.PREFERRED_SIZE, 132,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(825, Short.MAX_VALUE))
    );

    getContentPane().add(adminMenuPanel, "card6");

    jLabel188.setFont(new java.awt.Font("Cantarell", 1, 36)); // NOI18N
    jLabel188.setText("Driver");

    jComboBox26.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));
    jComboBox26.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jComboBox26ActionPerformed(evt);
        }
    });

    jLabel190.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
    jLabel190.setText("First Name:");

    jLabel191.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
    jLabel191.setText("Last Name:");

    jLabel192.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
    jLabel192.setText("AT:");

```

```

jLabel93.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel93.setText("Branch Code");

jLabel94.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel94.setText("Salary:");

jLabel96.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel96.setText("License:");

jLabel97.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel97.setText("Route:");

jLabel98.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel98.setText("Driver Exp:");

jLabel99.setText("dr_name");

jLabel100.setText("dr_lname");

jLabel101.setText("dr_at");

jLabel102.setText("dr_br_code");

jLabel103.setText("dr_salar");

jLabel104.setText("dr_lic");

jLabel105.setText("dr_route");

jLabel106.setText("dr_xp");

jLabel95.setFont(new java.awt.Font("Cantarell", 1, 24)); // NOI18N
jLabel95.setText("Trips");

jTable24.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null},
        {null, null, null},
        {null, null, null},
        {null, null, null}
    },
    new String [] {
        "Trip ID", "Start Date", "End Date"
    }
));
jTable24.getTableHeader().setReorderingAllowed(false);
jScrollPane25.setViewportView(jTable24);

javax.swing.GroupLayout driverMenuPanelLayout = new
javax.swing.GroupLayout(driverMenuPanel);
driverMenuPanel.setLayout(driverMenuPanelLayout);
driverMenuPanelLayout.setHorizontalGroup(

driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
    .addGroup(driverMenuPanelLayout.createSequentialGroup())

```

```

        .addGap(55, 55, 55)

        .addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
            lignment.TRAILING)
            .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
                driverMenuPanelLayout.createSequentialGroup())

        .addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
            lignment.TRAILING)
            .addComponent(jLabel192)
            .addComponent(jLabel193)
            .addComponent(jLabel191)
            .addComponent(jLabel190)
            .addComponent(jLabel194)
            .addComponent(jLabel196)
            .addComponent(jLabel197)
            .addComponent(jLabel198))
            .addGap(28, 28, 28)

        .addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
            lignment.LEADING, false)
            .addComponent(jLabel199,
                javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
                Short.MAX_VALUE)
            .addComponent(jLabel1100,
                javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
                Short.MAX_VALUE)
            .addComponent(jLabel1101,
                javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
                Short.MAX_VALUE)
            .addComponent(jLabel1102,
                javax.swing.GroupLayout.DEFAULT_SIZE, 115, Short.MAX_VALUE)
            .addComponent(jLabel1103,
                javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
                Short.MAX_VALUE)
            .addComponent(jLabel1104,
                javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
                Short.MAX_VALUE)
            .addComponent(jLabel1105,
                javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
                Short.MAX_VALUE)
            .addComponent(jLabel1106,
                javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
                Short.MAX_VALUE)))
            .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
                driverMenuPanelLayout.createSequentialGroup())
                .addComponent(jLabel188)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jComboBox26,
                javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
                javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(87, 87, 87)

        .addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
            lignment.LEADING)
            .addComponent(jScrollPane25,

```



```

javax.swing.GroupLayout.PREFERRED_SIZE, 575,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel195))
        .addContainerGap(291, Short.MAX_VALUE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
driverMenuPanelLayout.createSequentialGroup())
        .addContainerGap(55, Short.MAX_VALUE)
        .addComponent(jSeparator8,
javax.swing.GroupLayout.PREFERRED_SIZE, 1137,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(65, Short.MAX_VALUE))
    );
    driverMenuPanelLayout.setVerticalGroup(

driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
        .addGroup(driverMenuPanelLayout.createSequentialGroup())
        .addGap(25, 25, 25)

.addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING, false)
        .addComponent(jComboBox26)
        .addComponent(jLabel188,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        .addGap(18, 18, 18)
        .addComponent(jSeparator8,
javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
        .addComponent(jLabel190)
        .addComponent(jLabel199)
        .addComponent(jLabel195))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
        .addGroup(driverMenuPanelLayout.createSequentialGroup())

.addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
        .addComponent(jLabel191)
        .addComponent(jLabel1100))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
        .addComponent(jLabel192)
        .addComponent(jLabel1101))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

.addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                .addComponent(jLabel193)
                .addComponent(jLabel1102))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                .addComponent(jLabel194)
                .addComponent(jLabel1103))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
                .addComponent(jLabel196)
                .addComponent(jLabel1104))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                .addComponent(jLabel197)
                .addComponent(jLabel1105))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(driverMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
                .addComponent(jLabel198)
                .addComponent(jLabel1106))
                .addComponent(jScrollPane25,
javax.swing.GroupLayout.PREFERRED_SIZE, 241,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addContainerGap(733, Short.MAX_VALUE))
    );

getContentPane().add(driverMenuPanel, "card7");

GuideMenuPanel.setPreferredSize(new java.awt.Dimension(1250, 820));

jLabel1107.setFont(new java.awt.Font("Cantarell", 1, 36)); // NOI18N
jLabel1107.setText("Guide");

jComboBox27.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));
jComboBox27.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jComboBox27ActionPerformed(evt);
    }
});

jLabel1109.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel1109.setText("First Name:");

```

```

jLabel110.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel110.setText("Last Name:");

jLabel111.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel111.setText("AT:");

jLabel112.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel112.setText("Branch Code:");

jLabel113.setFont(new java.awt.Font("Cantarell", 1, 18)); // NOI18N
jLabel113.setText("Salary:");

jLabel114.setText("g_fn");

jLabel115.setText("g_ln");

jLabel116.setText("g_at");

jLabel117.setText("g-br");

jLabel118.setText("g_sal");

jTable25.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null},
        {null},
        {null},
        {null}
    },
    new String [] {
        "Languages"
    }
));
jScrollPane26.setViewportViewView(jTable25);

jTable26.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null},
        {null, null, null},
        {null, null, null},
        {null, null, null}
    },
    new String [] {
        "Trip ID", "Start Date", "End Date"
    }
));
jScrollPane27.setViewportViewView(jTable26);

jLabel119.setFont(new java.awt.Font("Cantarell", 1, 24)); // NOI18N
jLabel119.setText("Trips");

jTextArea2.setColumns(20);
jTextArea2.setRows(5);
jScrollPane28.setViewportViewView(jTextArea2);

jLabel120.setFont(new java.awt.Font("Cantarell", 1, 24)); // NOI18N
jLabel120.setText("CV");

```

```

        javax.swing.GroupLayout GuideMenuPanelLayout = new
javax.swing.GroupLayout (GuideMenuPanel) ;
        GuideMenuPanel.setLayout (GuideMenuPanelLayout) ;
        GuideMenuPanelLayout.setHorizontalGroup (

GuideMenuPanelLayout.createParallelGroup (javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup (GuideMenuPanelLayout.createSequentialGroup ()
            .addGap (56, 56, 56)

.addGroup (GuideMenuPanelLayout.createParallelGroup (javax.swing.GroupLayout.Alignment.LEADING)

            .addComponent (jLabel1120)

.addGroup (GuideMenuPanelLayout.createParallelGroup (javax.swing.GroupLayout.Alignment.LEADING, false)

            .addComponent (jScrollPane28)

.addGroup (GuideMenuPanelLayout.createSequentialGroup ()

.addGroup (GuideMenuPanelLayout.createParallelGroup (javax.swing.GroupLayout.Alignment.TRAILING, false)

.addGroup (GuideMenuPanelLayout.createSequentialGroup ()
            .addGap (0, 0, Short.MAX_VALUE)

.addGroup (GuideMenuPanelLayout.createParallelGroup (javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup (GuideMenuPanelLayout.createSequentialGroup ()

.addGroup (GuideMenuPanelLayout.createParallelGroup (javax.swing.GroupLayout.Alignment.TRAILING)

            .addComponent (jLabel1111)
            .addComponent (jLabel1112)
            .addComponent (jLabel1110)
            .addComponent (jLabel1109,

javax.swing.GroupLayout.PREFERRED_SIZE, 97,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addComponent (jLabel1113) )

.addGroup (GuideMenuPanelLayout.createParallelGroup (javax.swing.GroupLayout.Alignment.LEADING)

.addGroup (GuideMenuPanelLayout.createSequentialGroup ()
            .addGap (82, 82, 82)
            .addComponent (jLabel1114,

javax.swing.GroupLayout.PREFERRED_SIZE, 114,
javax.swing.GroupLayout.PREFERRED_SIZE) )

.addGroup (GuideMenuPanelLayout.createSequentialGroup ()
            .addGap (81, 81, 81)
            .addComponent (jLabel1115,

javax.swing.GroupLayout.PREFERRED_SIZE, 114,
javax.swing.GroupLayout.PREFERRED_SIZE) ) ) )

```

```

.addGroup(GuideMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                                                .addComponent(jLabel118,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 115, Short.MAX_VALUE)
                                                .addComponent(jLabel116,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                                                .addComponent(jLabel117,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)))
                                                .addGap(59, 59, 59))

.addGroup(javax.swing.GroupLayout.Alignment.LEADING,
GuideMenuPanelLayout.createSequentialGroup())
                                                .addComponent(jLabel1107)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                                                .addComponent(jComboBox27,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)))

.addGroup(GuideMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                                .addComponent(jScrollPane27,
javax.swing.GroupLayout.PREFERRED_SIZE, 574,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                                .addComponent(jLabel119)))
                                                .addComponent(jScrollPane26,
javax.swing.GroupLayout.PREFERRED_SIZE, 194,
javax.swing.GroupLayout.PREFERRED_SIZE))
                                                .addComponent(jSeparator9,
javax.swing.GroupLayout.PREFERRED_SIZE, 1137,
javax.swing.GroupLayout.PREFERRED_SIZE))
                                                .addContainerGap(64, Short.MAX_VALUE))
);
GuideMenuPanelLayout.setVerticalGroup(

GuideMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(GuideMenuPanelLayout.createSequentialGroup()
        .addGap(33, 33, 33)

.addGroup(GuideMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
        .addComponent(jComboBox27)
        .addComponent(jLabel1107,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        .addGap(18, 18, 18)
        .addComponent(jSeparator9,
javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(8, 8, 8)

```

```

.addGroup(GuideMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel1109)
    .addComponent(jLabel1114)
    .addComponent(jLabel1119))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(GuideMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(GuideMenuPanelLayout.createSequentialGroup())

.addGroup(GuideMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel1110)
    .addComponent(jLabel1115))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(GuideMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel1111)
    .addComponent(jLabel1116))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(GuideMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel1112)
    .addComponent(jLabel1117))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(GuideMenuPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel1113)
    .addComponent(jLabel1118))
    .addGap(18, 18, 18)
    .addComponent(jScrollPane26,
javax.swing.GroupLayout.PREFERRED_SIZE, 105,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addComponent(jScrollPane27,
javax.swing.GroupLayout.PREFERRED_SIZE, 257,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jLabel1120)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jScrollPane28,
javax.swing.GroupLayout.PREFERRED_SIZE, 194,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addContainerGap(469, Short.MAX_VALUE))
);

getContentPane().add(GuideMenuPanel, "card8");

```

```
jMenu1.setText("Data");
jMenu1.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        jMenu1MouseClicked(evt);
    }
});
jMenu1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenu1ActionPerformed(evt);
    }
});

jMenuItem2.setText("Admins");
jMenuItem2.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        jMenuItem2MouseClicked(evt);
    }
});
jMenuItem2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem2ActionPerformed(evt);
    }
});
jMenu1.add(jMenuItem2);

jMenuItem1.setText("Branches");
jMenuItem1.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        jMenuItem1MouseClicked(evt);
    }
});
jMenuItem1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem1ActionPerformed(evt);
    }
});
jMenu1.add(jMenuItem1);

jMenuItem3.setText("Destinations");
jMenuItem3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem3ActionPerformed(evt);
    }
});
jMenu1.add(jMenuItem3);

jMenuItem4.setText("Drivers");
jMenuItem4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem4ActionPerformed(evt);
    }
});
jMenu1.add(jMenuItem4);

jMenuItem5.setText("Events");
jMenuItem5.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```

        jMenuItem5ActionPerformed(evt);
    }
});
jMenu1.add(jMenuItem5);

jMenuItem6.setText("Guides");
jMenuItem6.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem6ActionPerformed(evt);
    }
});
jMenu1.add(jMenuItem6);

jMenuItem7.setText("It Officers");
jMenuItem7.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem7ActionPerformed(evt);
    }
});
jMenu1.add(jMenuItem7);

jMenuItem8.setText("Languages");
jMenuItem8.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem8ActionPerformed(evt);
    }
});
jMenu1.add(jMenuItem8);

jMenuItem9.setText("Managers");
jMenuItem9.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem9ActionPerformed(evt);
    }
});
jMenu1.add(jMenuItem9);

jMenuItem10.setText("Offers");
jMenuItem10.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem10ActionPerformed(evt);
    }
});
jMenu1.add(jMenuItem10);

jMenuItem11.setText("Phones");
jMenuItem11.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem11ActionPerformed(evt);
    }
});
jMenu1.add(jMenuItem11);

jMenuItem12.setText("Reservations");
jMenuItem12.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem12ActionPerformed(evt);
    }
});
jMenu1.add(jMenuItem12);

```



```

    }
    });
    jMenu1.add(jMenuItem12);

    jMenuItem13.setText("Reservation offers");
    jMenuItem13.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem13ActionPerformed(evt);
        }
    });
    jMenu1.add(jMenuItem13);

    jMenuItem14.setText("Trips");
    jMenuItem14.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem14ActionPerformed(evt);
        }
    });
    jMenu1.add(jMenuItem14);

    jMenuItem15.setText("Workers");
    jMenuItem15.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem15ActionPerformed(evt);
        }
    });
    jMenu1.add(jMenuItem15);

    jMenuItem19.setText("TravelTo");
    jMenuItem19.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem19ActionPerformed(evt);
        }
    });
    jMenu1.add(jMenuItem19);

    jMenuBar1.add(jMenu1);

    jMenuItem2.setText("Branches");
    jMenuItem2.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem2ActionPerformed(evt);
        }
    });

    jMenuItem17.setText("Branch Info ");
    jMenuItem17.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem17ActionPerformed(evt);
        }
    });
    jMenuItem2.add(jMenuItem17);

    jMenuItem23.setText("Available Trips");
    jMenuItem23.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem23ActionPerformed(evt);
        }
    });

```

```

    }
    });
    jMenu2.add(jMenuItem23);

    jMenuBar1.add(jMenu2);

    jMenu3.setText("Offers");
    jMenu3.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenu3ActionPerformed(evt);
        }
    });

    jMenuItem16.setText("Offers Procedure");
    jMenuItem16.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem16ActionPerformed(evt);
        }
    });
    jMenu3.add(jMenuItem16);

    jMenuBar1.add(jMenu3);

    jMenu4.setText("Logs");
    jMenu4.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenu4ActionPerformed(evt);
        }
    });

    jMenuItem18.setText("Logs data");
    jMenuItem18.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem18ActionPerformed(evt);
        }
    });
    jMenu4.add(jMenuItem18);

    jMenuBar1.add(jMenu4);

    jMenu5.setText("Workers");

    jMenuItem20.setText("Admin Card");
    jMenuItem20.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem20ActionPerformed(evt);
        }
    });
    jMenu5.add(jMenuItem20);

    jMenuItem21.setText("Driver Card");
    jMenuItem21.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem21ActionPerformed(evt);
        }
    });
    jMenu5.add(jMenuItem21);

```

```

jMenuItem22.setText("Guide Card");
jMenuItem22.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem22ActionPerformed(evt);
    }
});
jMenu5.add(jMenuItem22);

jMenuBar1.add(jMenu5);

jMenu6.setText("Log Out");
jMenu6.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mousePressed(java.awt.event.MouseEvent evt) {
        jMenu6MousePressed(evt);
    }
});
jMenuBar1.add(jMenu6);

setJMenuBar(jMenuBar1);

pack();
} // </editor-fold> // GEN-END: initComponents

private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_jMenuItem2ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(adminsPanel).setVisible(true);

    jComboBox4.removeAllItems();
    jComboBox5.removeAllItems();

    try {
        Connection con = ConnectionToDb.connectMySQLDB();
        Statement stmt = con.createStatement();
        ResultSet rs = stmt.executeQuery("select * from admin");
        DefaultTableModel tbModel =
(DefaultTableModel) jTable8.getModel();
        tbModel.setRowCount(0);
        while(rs.next()) {

            String adm_AT = rs.getString("adm_AT");
            String adm_type = rs.getString("adm_type");
            String adm_diploma = rs.getString("adm_diploma");

            String tbData[] = {adm_AT, adm_type, adm_diploma};

            tbModel.addRow(tbData);
        }
    }
}

```

```

        rs=stmt.executeQuery("select distinct wrk_AT from worker");
        while(rs.next()){
            String br_code = rs.getString("wrk_AT");
            jComboBox4.addItem(br_code);
        }

        rs=stmt.executeQuery("select distinct adm_type from admin");
        while(rs.next()){
            String admType = rs.getString("adm_type");
            jComboBox5.addItem(admType);
        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
        JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem2ActionPerformed

private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem1ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(branchesPanel).setVisible(true);

    try {
        Connection con = ConnectionToDb.connectMySqlDB();
        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from branch");
        DefaultTableModel tbModel =
(DefaultTableModel) jTable2.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String br_code = String.valueOf(rs.getInt("br_code"));
            String br_num = String.valueOf(rs.getInt("br_num"));
            String br_street = rs.getString("br_street");
            String br_city = rs.getString("br_city");

            String tbData[] = {br_code, br_num, br_street, br_city};
            //DefaultTableModel tbModel =
(DefaultTableModel) jTable2.getModel();

            tbModel.addRow(tbData);
        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();

```

```

JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem1ActionPerformed

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton1ActionPerformed
    // TODO add your handling code here:

    if( !jTextField2.getText().equals("") &&
!jTextField3.getText().equals("") && !jTextField4.getText().equals("") &&
!jTextField5.getText().equals("") ){
        String br_code = jTextField2.getText();
        String br_num = jTextField3.getText();
        String br_street = jTextField4.getText();
        String br_city = jTextField5.getText();

        String insertBranchCommand = "INSERT INTO branch VALUES (" +
br_code + "," + br_num + " , " + br_street + " , " + br_city + ")";

        try {
            Connection con = ConnectionToDb.connectMySQLDB();
            Statement stmt=con.createStatement();
            stmt.executeUpdate(insertBranchCommand);

            DefaultTableModel tbModel =
(DefaultTableModel)jTable2.getModel();
            String tbData[] = { br_code, br_num, br_street , br_city};
            tbModel.addRow(tbData);
            con.close();
        }
        catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
    }

} //GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton2ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable2.getSelectedRow();
    String br_code = jTable2.getValueAt(selectedRow, 0).toString();
    System.out.println(br_code);
    String deleteBranchCommand = "delete from branch where br_code= " +
br_code + ";";

    try {
        Connection con = ConnectionToDb.connectMySQLDB();

        Statement stmt=con.createStatement();
        stmt.executeUpdate(deleteBranchCommand);

        DefaultTableModel tbModel =
(DefaultTableModel)jTable2.getModel();

```

```

        tbModel.removeRow(selectedRow);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

    }//GEN-LAST:event_jButton2ActionPerformed

    private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jMenuItem1ActionPerformed

    }//GEN-LAST:event_jMenuItem1ActionPerformed

    private void jMenuItem1MouseClicked(java.awt.event.MouseEvent evt) { //GEN-FIRST:event_jMenuItem1MouseClicked

    }//GEN-LAST:event_jMenuItem1MouseClicked

    private void jMenuItem2MouseClicked(java.awt.event.MouseEvent evt) { //GEN-FIRST:event_jMenuItem2MouseClicked
        // TODO add your handling code here:
    }//GEN-LAST:event_jMenuItem2MouseClicked

    private void jMenuItem3ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jMenuItem3ActionPerformed
        // TODO add your handling code here:
        dataMenuPanel.setVisible(true);
        branchesMenuPanel.setVisible(false);
        offersMenuPanel.setVisible(false);
        logsMenuPanel.setVisible(false);
        adminMenuPanel.setVisible(false);
        driverMenuPanel.setVisible(false);
        GuideMenuPanel.setVisible(false);
        dataMenuPanel.removeAll();
        dataMenuPanel.add(destinationsPanel).setVisible(true);

        jComboBox20.removeAllItems();

    try {
        Connection con = ConnectionToDb.connectMySQLDB();

        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from destination");
        DefaultTableModel tbModel =
(DefaultTableModel)jTable3.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String dst_id = String.valueOf(rs.getInt("dst_id"));
            String dst_name = rs.getString("dst_name");
            String dst_descr = rs.getString("dst_descr");

```

```
String dst_rtype = rs.getString("dst_rtype");
String dst_language = rs.getString("dst_language");
String dst_location =
String.valueOf(rs.getInt("dst_location") );
String tbData[] = {dst_id, dst_name, dst_descr,
dst_rtype, dst_language, dst_location};
//DefaultTableModel tbModel =
(DefaultTableModel) jTable2.getModel();

tbModel.addRow(tbData);
}

rs = stmt.executeQuery("select distinct dst_rtype from
destination");
while(rs.next()){
String dst = rs.getString("dst_rtype");
jComboBox20.addItem(dst);
}

con.close();
}
catch(Exception e){JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem3ActionPerformed

private void jMenuItem4ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem4ActionPerformed
// TODO add your handling code here:
dataMenuPanel.setVisible(true);
branchesMenuPanel.setVisible(false);
offersMenuPanel.setVisible(false);
logsMenuPanel.setVisible(false);
adminMenuPanel.setVisible(false);
driverMenuPanel.setVisible(false);
GuideMenuPanel.setVisible(false);
dataMenuPanel.removeAll();
dataMenuPanel.add(driversPanel).setVisible(true);

jComboBox6.removeAllItems();
jComboBox21.removeAllItems();
jComboBox22.removeAllItems();

try {
Connection con = ConnectionToDb.connectMySqlDB();

Statement stmt=con.createStatement();
ResultSet rs=stmt.executeQuery("select * from driver");
DefaultTableModel tbModel =
(DefaultTableModel) jTable7.getModel();
tbModel.setRowCount(0);
while(rs.next()){

String drv AT = rs.getString("drv AT");
```

```

        String drv_license = rs.getString("drv_license");
        String drv_route = rs.getString("drv_route");
        String drv_experience =
String.valueOf(rs.getInt("drv_experience") );

        String tbData[] = {drv_AT ,drv_license , drv_route,
drv_experience};

        tbModel.addRow(tbData);
    }

    rs=stmt.executeQuery("select distinct wrk_AT from worker");
    while(rs.next()){

        //String br_code = String.valueOf(rs.getInt("br_code") );
        String br_code = rs.getString("wrk_AT");
        jComboBox6.addItem(br_code);

    }

    rs=stmt.executeQuery("select distinct drv_license from driver
order by drv_license asc");
    while(rs.next()){

        //String br_code = String.valueOf(rs.getInt("br_code") );
        String license = rs.getString("drv_license");
        jComboBox21.addItem(license);

    }

    rs=stmt.executeQuery("select distinct drv_route from driver");
    while(rs.next()){

        //String br_code = String.valueOf(rs.getInt("br_code") );
        String route = rs.getString("drv_route");
        jComboBox22.addItem(route);

    }

    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem4ActionPerformed

private void jMenuItem5ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem5ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();

```



```

dataMenuPanel.add(eventsPanel).setVisible(true);
jComboBox15.removeAllItems();

try {
    Connection con = ConnectionToDb.connectMySQLDB();

    Statement stmt=con.createStatement();
    ResultSet rs=stmt.executeQuery("select * from event");
    DefaultTableModel tbModel =
(DefaultTableModel) jTable11.getModel();
    tbModel.setRowCount(0);
    while(rs.next()){

        String ev_tr_id = String.valueOf(rs.getInt("ev_tr_id"));
        String ev_start = rs.getDate("ev_start").toString();
        String ev_end = rs.getDate("ev_end").toString();
        String ev_descr = rs.getString("ev_descr");

        String tbData[] = {ev_tr_id, ev_start, ev_end, ev_descr};

        tbModel.addRow(tbData);
    }

    rs=stmt.executeQuery("select distinct tr_id from trip");
    while(rs.next()){

        String tr_id = rs.getString("tr_id");
        jComboBox15.addItem(tr_id);
    }
    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem5ActionPerformed

private void jMenuItem6ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem6ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(guidesPanel).setVisible(true);
    jComboBox7.removeAllItems();
    try {
        Connection con = ConnectionToDb.connectMySQLDB();

        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from guide");
        DefaultTableModel tbModel =
(DefaultTableModel) jTable5.getModel();

```

```

        tbModel.setRowCount(0);
        while(rs.next()){

            String gui_AT = rs.getString("gui_AT");
            String gui_cv = rs.getString("gui_CV");

            String tbData[] = {gui_AT, gui_cv};

            tbModel.addRow(tbData);
        }

        rs=stmt.executeQuery("select distinct wrk_AT from worker");
        while(rs.next()){

            //String br_code = String.valueOf(rs.getInt("br_code") );
            String wrk_AT = rs.getString("wrk_AT");
            jComboBox7.addItem(wrk_AT);

        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
    JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem6ActionPerformed

private void jMenuItem7ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem7ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(itOfficersPanel).setVisible(true);

    jComboBox29.removeAllItems();
    jComboBox3.removeAllItems();
    try {
        Connection con = ConnectionToDb.connectMySQLDB();

        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from itOfficer");
        DefaultTableModel tbModel =
(DefaultTableModel) jTable15.getModel();
        tbModel.setRowCount(0);
        jComboBox17.removeAllItems();

        while(rs.next()){

            String it_AT = rs.getString("it_AT");
            String password = rs.getString("password");

```

```

        String start_date = rs.getString("start_date");
        String end_date = rs.getString("end_date");

        String tbData[] = { it_AT , password , start_date ,
end_date };

        tbModel.addRow(tbData);
    }

    rs=stmt.executeQuery("select distinct wrk_AT from worker");
    while(rs.next()){

        String wrk_AT = rs.getString("wrk_AT");
        jComboBox17.addItem(wrk_AT);

    }

    rs = stmt.executeQuery("select distinct br_code from
branch");
    while(rs.next()){
        String brCode = rs.getString("br_code");
        jComboBox29.addItem(brCode);
    }
    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem7ActionPerformed

private void jMenuItem8ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem8ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(languagesPanel).setVisible(true);

    jComboBox10.removeAllItems();

    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from languages");
        DefaultTableModel tbModel =
(DefaultTableModel)jTable6.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String lng_gui_AT = rs.getString("lng_gui_AT");
            String lng_language = rs.getString("lng_language");

```

```

        String tbData[] = {lng_gui_AT, lng_language};

        tbModel.addRow(tbData);
    }

    rs=stmt.executeQuery("select DISTINCT gui_AT from guide");
    while(rs.next()){

        //String br_code = String.valueOf(rs.getInt("br_code") );
        String gui_AT = rs.getString("gui_AT");
        jComboBox10.addItem(gui_AT);

    }

    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem8ActionPerformed

private void jMenuItem9ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem9ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(managersPanel).setVisible(true);

    jComboBox8.removeAllItems();
    jComboBox9.removeAllItems();

    try {
        Connection con = ConnectionToDb.connectMySQLDB();

        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from manages");
        DefaultTableModel tbModel =
(DefaultTableModel) jTable9.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String mng_adm_AT = rs.getString("mng_adm_AT");
            String mng_br_code = String.valueOf(rs.getInt("mng_br_code")
);

            String tbData[] = {mng_adm_AT , mng_br_code};

            tbModel.addRow(tbData);

        }
    }
}

```

```

        rs=stmt.executeQuery("select distinct adm_AT from admin");
        while(rs.next()){

            //String br_code = String.valueOf(rs.getInt("br_code") );
            String br_code = rs.getString("adm_AT");
            jComboBox8.addItem(br_code);

        }

        rs=stmt.executeQuery("select br_code from branch");
        while(rs.next()){

            //String br_code = String.valueOf(rs.getInt("br_code") );
            String br_code = rs.getString("br_code");
            jComboBox9.addItem(br_code);

        }

        con.close();
    }
    catch(Exception e){JFrame f=new JFrame();
    JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem9ActionPerformed

private void jMenuItem10ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem10ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(offersPanel).setVisible(true);

    jComboBox16.removeAllItems();

    try {
        Connection con = ConnectionToDb.connectMySQLDB();

        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from offers");
        DefaultTableModel tbModel =
(DefaultTableModel) jTable13.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String offer_id = String.valueOf( rs.getInt("offer_id") );
            String offer_startDate = rs.getString("offer_startDate");
            String offer_endDate = rs.getString("offer_endDate");
            String cost_per_person =
String.valueOf(rs.getFloat("cost_per_person") );
            String offer_dst_id = String.valueOf(
rs.getInt("offer_dst_id") );

```

```

        String tbData[] = {offer_id, offer_startDate, offer_endDate,
cost_per_person ,offer_dst_id};

        tbModel.addRow(tbData);
    }

    rs=stmt.executeQuery("select distinct dst_id from destination");
    while(rs.next()){

        String dst_id = rs.getString("dst_id");
        jComboBox16.addItem(dst_id);

    }
    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem10ActionPerformed

private void jMenuItem11ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem11ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(phonesPanel).setVisible(true);

    jComboBox3.removeAllItems();
    try {
        Connection con = ConnectionToDb.connectMySQLDB();

        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from phones");
        DefaultTableModel tbModel =
(DefaultTableModel)jTable1.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String ph_br_code =
String.valueOf(rs.getInt("ph_br_code"));
            String ph_number = rs.getString("ph_number");

            String tbData[] = {ph_br_code, ph_number};

            tbModel.addRow(tbData);
        }

        rs=stmt.executeQuery("select br_code from branch");
        while(rs.next()){

```

```

        //String br_code = String.valueOf(rs.getInt("br_code"));
        String br_code = rs.getString("br_code");
        jComboBox3.addItem(br_code);

    }

    con.close();
}

catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}


} //GEN-LAST:event_jMenuItem11ActionPerformed

private void jMenuItem12ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem12ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(reservationsPanel).setVisible(true);

    jComboBox14.removeAllItems();
    jComboBox19.removeAllItems();
    try {
        Connection con = ConnectionToDb.connectMySQLDB();

        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from reservation");
        DefaultTableModel tbModel =
(DefaultTableModel) jTable12.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String res_tr_id = String.valueOf(rs.getInt("res_tr_id"));
            String res_seatnum = String.valueOf(rs.getInt("res_seatnum"));

            String res_name = rs.getString("res_name");
            String res_lname = rs.getString("res_lname");
            String res_isadult = rs.getString("res_isadult");

            String tbData[] = {res_tr_id, res_seatnum, res_name,
res_lname,res_isadult};

            tbModel.addRow(tbData);

        }

        rs=stmt.executeQuery("select distinct tr_id from trip");
        while(rs.next()){

```

```

        String res_tr_id = rs.getString("tr_id");
        jComboBox14.addItem(res_tr_id);

    }
    rs=stmt.executeQuery("select distinct res_isadult from
reservation");
    while(rs.next()){

        String isAdult = rs.getString("res_isadult");
        jComboBox19.addItem(isAdult);

    }

    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem12ActionPerformed

private void jMenuItem13ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem13ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(reservationOffersPanel).setVisible(true);

    jComboBox18.removeAllItems();

    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from
reservation_offers");
        DefaultTableModel tbModel =
(DefaultTableModel) jTable14.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String res_offer_id =
String.valueOf(rs.getInt("res_offer_id"));
            String res_name = rs.getString("res_name");
            String res_lname = rs.getString("res_lname");
            String offer_id_trip = String.valueOf(
rs.getInt("offer_id_trip"));
            String res_deposit = String.valueOf(
rs.getFloat("res_deposit"));

            String tbData[] = {res_offer_id, res_name, res_lname,

```



```

offer_id_trip, res_deposit});

        tbModel.addRow(tbData);
    }

    rs=stmt.executeQuery("select distinct offer_id from offers");
    while(rs.next()){

        String br_code = rs.getString("offer_id");
        jComboBox18.addItem(br_code);

    }

    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem13ActionPerformed

private void jMenuItem14ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem14ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(tripsPanel).setVisible(true);

    jComboBox11.removeAllItems();
    jComboBox12.removeAllItems();
    jComboBox13.removeAllItems();

    try {
        Connection con = ConnectionToDb.connectMySQLDB();

        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from trip");
        DefaultTableModel tbModel =
(DefaultTableModel)jTable10.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String tr_id = String.valueOf(rs.getInt("tr_id") );
            String tr_departure =
String.valueOf(rs.getDate("tr_departure") );
            String tr_return = String.valueOf( rs.getDate("tr_return") );
            String tr_maxseats = String.valueOf(rs.getInt("tr_maxseats")
);

            String tr_cost = String.valueOf(rs.getFloat("tr_cost") );

            String tr_br_code = String.valueOf(rs.getInt("tr_br_code") );

```

```

        String tr_gui_AT = rs.getString("tr_gui_AT");
        String tr_drv_AT = rs.getString("tr_drv_AT");

        String tbData[] = {tr_id , tr_departure, tr_return,
tr_maxseats, tr_cost, tr_br_code, tr_gui_AT, tr_drv_AT};

        tbModel.addRow(tbData);
    }

    rs=stmt.executeQuery("select br_code from branch");
    while(rs.next()){

        String br_code = rs.getString("br_code");
        jComboBox11.addItem(br_code);
    }

    rs=stmt.executeQuery("select distinct gui_AT from guide");
    while(rs.next()){

        String gui_AT = rs.getString("gui_AT");
        jComboBox12.addItem(gui_AT);
    }

    rs=stmt.executeQuery("select drv_AT from driver");
    while(rs.next()){

        String drv_AT = rs.getString("drv_AT");
        jComboBox13.addItem(drv_AT);
    }

    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem14ActionPerformed

private void jMenuItem15ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem15ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();
    dataMenuPanel.add(workersPanel).setVisible(true);
    jComboBox2.removeAllItems();

    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();

```

```

        ResultSet rs=stmt.executeQuery("select * from worker");
        DefaultTableModel tbModel =
(DefaultTableModel)jTable4.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String wrk_AT = rs.getString("wrk_AT");
            String wrk_name = rs.getString("wrk_name");
            String wrk_lame = rs.getString("wrk_lame");
            String wrk_salary = String.valueOf(rs.getFloat("wrk_salary")
);
            String wrk_br_code = String.valueOf( rs.getInt("wrk_br_code")
);

            String tbData[] = {wrk_AT,wrk_name, wrk_lame, wrk_salary,
wrk_br_code};

            tbModel.addRow(tbData);

        }

        rs=stmt.executeQuery("select br_code from branch");
        while(rs.next()){

            //String br_code = String.valueOf(rs.getInt("br_code") );
            String br_code = rs.getString("br_code");
            jComboBox2.addItem(br_code);

        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem15ActionPerformed

private void jMenuItem17ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem17ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(false);
    branchesMenuPanel.setVisible(true);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    branchesMenuPanel.removeAll();
    branchesMenuPanel.add(branchInfoPanel).setVisible(true);
    jComboBox1.removeAllItems();
    try {
        Connection con = ConnectionToDb.connectMySQLDB();

        /*Branch*/
        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select br_code from branch");

```

```

        //Remove all items from combobox

        while(rs.next()){

            //String br_code = String.valueOf(rs.getInt("br_code") );
            String br_code = rs.getString("br_code");
            jComboBox1.addItem(br_code);

        }
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
    JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem17ActionPerformed

private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jMenuItem2ActionPerformed
    // TODO add your handling code here:

} //GEN-LAST:event_jMenuItem2ActionPerformed

private void jMenuItem3ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jMenuItem3ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jMenuItem3ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton4ActionPerformed
    // TODO add your handling code here:

    String lastName = jTextField1.getText();

    String procedureShowAvailableTrips = "call data_last_name(\"" +
lastName + "\" ) ";

    try {
        Connection con = ConnectionToDb.connectMySQLDB();

        DefaultTableModel tbModelData =
(DefaultTableModel)jTable21.getModel();
        tbModelData.setRowCount(0);

        DefaultTableModel tbModelData2 =
(DefaultTableModel)jTable22.getModel();
        tbModelData2.setRowCount(0);

        CallableStatement stmt =
con.prepareCall(procedureShowAvailableTrips);
        boolean results = stmt.execute();

        ResultSet rs = stmt.getResultSet();

        while(rs.next()){
            String res_name = rs.getString("res_name");

```

```

        String res_lname = rs.getString("res_lname");
        String offer_id = rs.getString("offer_id");
        String offer_startDate = rs.getString("offer_startDate");
        String offer_endDate = rs.getString("offer_endDate");
        String cost_per_person = rs.getString("cost_per_person");
        String offer_dst_id = rs.getString("offer_dst_id");

        String tbData[] = {res_name , res_lname , offer_id
,offer_startDate ,offer_endDate ,cost_per_person ,offer_dst_id };

        tbModelData.addRow(tbData);
    }

    results = stmt.getMoreResults();
    if(results == true){
        rs = stmt.getResultSet();
        while(rs.next()){
            String offer_id = rs.getString("offer_id");
            String offer_startDate = rs.getString("offer_startDate");
            String offer_endDate = rs.getString("offer_endDate");
            String cost_per_person = rs.getString("cost_per_person");
            String offer_dst_id = rs.getString("offer_dst_id");
            String count_res_lname =
rs.getString("count(res_lname)");

            String tbData[] = {offer_id , offer_startDate
,offer_endDate ,cost_per_person ,offer_dst_id ,count_res_lname };

            tbModelData2.addRow(tbData);
        }
    }

    con.close();
}
catch(Exception e){JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jButton4ActionPerformed

private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jTextField1ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jTextField1ActionPerformed

private void jMenuItem16ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem16ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(false);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(true);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);

} //GEN-LAST:event_jMenuItem16ActionPerformed

```

```

        private void jTextField2ActionPerformed(java.awt.event.ActionEvent evt)
        { //GEN-FIRST:event_jTextField2ActionPerformed
            // TODO add your handling code here:
        } //GEN-LAST:event_jTextField2ActionPerformed

        private void jTextField4ActionPerformed(java.awt.event.ActionEvent evt)
        { //GEN-FIRST:event_jTextField4ActionPerformed
            // TODO add your handling code here:
        } //GEN-LAST:event_jTextField4ActionPerformed

        private void jButton5ActionPerformed(java.awt.event.ActionEvent evt)
        { //GEN-FIRST:event_jButton5ActionPerformed
            // TODO add your handling code here:

            int selectedRow = jTable2.getSelectedRow();
            if(selectedRow != -1){

                String br_code = jTable2.getValueAt(selectedRow, 0).toString();

                try {

                    Connection con = ConnectionToDb.connectMySQLDB();

                    String[] data = new String[ 4 ];

                    data[0] = jTextField2.getText();
                    data[1] = jTextField3.getText();
                    data[2] = jTextField4.getText();
                    data[3] = jTextField5.getText();

                    String command0 = "UPDATE branch set br_code=" + data[0] + "
where br_code= " + br_code + " ";
                    String command1 = "UPDATE branch set br_num=" + data[1] + "
where br_code= " + br_code + " ";
                    String command2 = "UPDATE branch set br_street='" + data[2] + "
'" where br_code= " + br_code + " ";
                    String command3 = "UPDATE branch set br_city='" + data[3] + "
'" where br_code= " + br_code + " ";
                    System.out.println(command0);
                    System.out.println(command1);
                    System.out.println(command2);
                    System.out.println(command3);

                    if(!data[1].equals("")){
                        CallableStatement stmt = con.prepareCall(command1);
                        stmt.executeUpdate();
                    }
                    if(!data[2].equals("")){
                        CallableStatement stmt = con.prepareCall(command2);
                        stmt.executeUpdate();
                    }
                    if(!data[3].equals("")){
                        CallableStatement stmt = con.prepareCall(command3);
                        stmt.executeUpdate();
                    }
                    if(!data[0].equals("")){

```

```

        CallableStatement stmt = con.prepareCall(command0);
        stmt.executeUpdate();

    }

    System.out.println(br_code);
    //Update table
    CallableStatement stmt = con.prepareCall("select * from
branch");

    ResultSet rs= stmt.executeQuery();
    DefaultTableModel tbModel =
(DefaultTableModel) jTable2.getModel();
    tbModel.setRowCount(0);
    while(rs.next()){

        String br_code2 = rs.getString("br_code") ;
        String br_num = rs.getString("br_num") ;
        String br_street = rs.getString("br_street");
        String br_city = rs.getString("br_city");

        String tbData[] = {br_code2, br_num, br_street, br_city};
        //DefaultTableModel tbModel =
(DefaultTableModel) jTable2.getModel();

        tbModel.addRow(tbData);
    }

    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
}

} //GEN-LAST:event_jButton5ActionPerformed

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton6ActionPerformed
    // TODO add your handling code here:

    if( !jTextField6.getText().equals("") &&
!jTextField7.getText().equals("") && !jTextField8.getText().equals("") &&
!jTextField9.getText().equals("") ){
        String wrk_at = jTextField6.getText();
        String wrk_name = jTextField7.getText();
        String wrk_lname = jTextField8.getText();
        String wrk_salary = jTextField9.getText();
        String wrk_br_code = jComboBox2.getSelectedItem().toString();

        String insertWorkerCommand = "INSERT INTO worker VALUES ('" +
wrk_at + "','" + wrk_name + "','" + wrk_lname + "','" + wrk_salary + "','" +
wrk_br_code + "');" ;

        try {
            Connection con = ConnectionToDb.connectMySQLDB();

```

```

        Statement stmt=con.createStatement();
        stmt.executeUpdate(insertWorkerCommand);

        DefaultTableModel tbModel =
(DefaultTableModel)jTable4.getModel();
        String tbData[] = { wrk_at, wrk_name, wrk_lname , wrk_salary,
wrk_br_code };
        tbModel.addRow(tbData);
        con.close();
    }
    catch(SQLException e){
        JFrame f=new JFrame();

JOptionPane.showMessageDialog(f,e.getMessage(),"Alert",JOptionPane.WARNING_ME
SSAGE);
    }
}

} //GEN-LAST:event_jButton6ActionPerformed

private void jButton7ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton7ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable4.getSelectedRow();
    if(selectedRow != -1){

        String key1 = jTable4.getValueAt(selectedRow, 0).toString();
        String key2 = jTable4.getValueAt(selectedRow, 4).toString();

        try {

            Connection con = ConnectionToDb.connectMySqlDB();

            String[] data = new String[ 5 ];

            data[0] = jTextField6.getText();
            data[1] = jTextField7.getText();
            data[2] = jTextField8.getText();
            data[3] = jTextField9.getText();
            data[4] = jComboBox2.getSelectedItem().toString();

            String command1 = "UPDATE worker set wrk_name = '" +
data[1] + "' where wrk_AT = '" + key1 + "' and wrk_br_code = '"+ key2 +" ";
            String command2 = "UPDATE worker set wrk_lame = '" +
data[2] + "' where wrk_AT = '" + key1 + "' and wrk_br_code = '"+ key2 +" ";
            String command3 = "UPDATE worker set wrk_salary = " +
data[3] + " where wrk_AT = '" + key1 + "' and wrk_br_code = '"+ key2 +" ";
            String command4 = "UPDATE worker set wrk_br_code = " +
data[4] + " where wrk_AT = '" + key1 + "' and wrk_br_code = '"+ key2 +" ";

            if(!data[1].equals("")){
                CallableStatement stmt = con.prepareCall(command1);
                stmt.executeUpdate();
            }
        }
    }
}

```



```

    }
    if(!data[2].equals("")){
        CallableStatement stmt = con.prepareCall(command2);
        stmt.executeUpdate();
    }
    if(!data[3].equals("")){
        CallableStatement stmt = con.prepareCall(command3);
        stmt.executeUpdate();
    }
    if(!data[4].equals("")){
        CallableStatement stmt = con.prepareCall(command4);
        stmt.executeUpdate();
        key2 = data[4];
    }

    if(!data[0].equals("")){
        String command0 = "UPDATE worker set wrk_AT = '" +
data[0] + "' where wrk_AT = '" + key1 + "' and wrk_br_code = '" + key2 + "' ";
        CallableStatement stmt = con.prepareCall(command0);
        stmt.executeUpdate();
    }

    //Update table
    CallableStatement stmt = con.prepareCall("select * from
worker");

    ResultSet rs= stmt.executeQuery();
    DefaultTableModel tbModel =
(DefaultTableModel)jTable4.getModel();
    tbModel.setRowCount(0);
    while(rs.next()){

        String wrk_AT = rs.getString("wrk_AT") ;
        String wrk_name = rs.getString("wrk_name") ;
        String wrk_lame = rs.getString("wrk_lame") ;
        String wrk_salary = rs.getString("wrk_salary") ;
        String wrk_br_code = rs.getString("wrk_br_code") ;

        String tbData[] = {wrk_AT, wrk_name ,wrk_lame
,wrk_salary , wrk_br_code};

        tbModel.addRow(tbData);
    }

    con.close();
}
catch(SQLException e){
    JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e.getMessage(),"Alert",JOptionPane.WARNING_ME
SSAGE);
}
}

```

```

    } //GEN-LAST:event_jButton7ActionPerformed

    private void jTextField7ActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_jTextField7ActionPerformed
        // TODO add your handling code here:
    } //GEN-LAST:event_jTextField7ActionPerformed

    private void jComboBox7ActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_jComboBox7ActionPerformed
        // TODO add your handling code here:
    } //GEN-LAST:event_jComboBox7ActionPerformed

    private void jButton36ActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_jButton36ActionPerformed
        // TODO add your handling code here:

        if( !jTextField26.getText().equals("") &&
!jTextField27.getText().equals("") && !jTextField28.getText().equals("") &&
!jTextField29.getText().equals("")
        && !jTextField30.getText().equals("") ){
            String tr_id = jTextField26.getText();
            String tr_departure = jTextField27.getText();
            String tr_return = jTextField28.getText();
            String tr_maxseats = jTextField29.getText();
            String tr_cost = jTextField30.getText();
            String tr_br_code = jComboBox11.getSelectedItem().toString();
            String tr_gui_AT = jComboBox12.getSelectedItem().toString();
            String tr_drv_AT = jComboBox13.getSelectedItem().toString();

            String insertTripCommand = "INSERT INTO trip VALUES (" + tr_id +
",'" + tr_departure + "','" + tr_return + "','" + tr_maxseats + "','" + tr_cost
+ "','" + tr_br_code + "','" + tr_gui_AT + "','" + tr_drv_AT + "')";

            try {
                Connection con = ConnectionToDb.connectMySqlDB();

                Statement stmt=con.createStatement();
                stmt.executeUpdate(insertTripCommand);

                DefaultTableModel tbModel =
(DefaultTableModel)jTable10.getModel();
                String tbData[] = { tr_id , tr_departure , tr_return ,
tr_maxseats , tr_cost , tr_br_code , tr_gui_AT , tr_drv_AT };
                tbModel.addRow(tbData);
                con.close();
            }

            catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

        }

    } //GEN-LAST:event_jButton36ActionPerformed

    private void jButton39ActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_jButton39ActionPerformed
        // TODO add your handling code here:

```

```

        if( !jTextField16.getText().equals("") &&
!jTextField35.getText().equals("") && !jTextField36.getText().equals("") &&
!jTextField37.getText().equals("") ){
            String offer_id = jTextField16.getText();
            String offer_startDate = jTextField35.getText();
            String offer_endDate = jTextField36.getText();
            String cost_per_person = jTextField37.getText();
            String offer_dst_id = jComboBox16.getSelectedItem().toString();

            String insertOfferCommand = "INSERT INTO offers VALUES (" +
offer_id + ", '" + offer_startDate + "', '" + offer_endDate + "', " +
cost_per_person + ", " + offer_dst_id + ")";

            try {
                Connection con = ConnectionToDb.connectMySqlDB();

                Statement stmt=con.createStatement();
                stmt.executeUpdate(insertOfferCommand);

                DefaultTableModel tbModel =
(DefaultTableModel)jTable13.getModel();
                String tbData[] = { offer_id , offer_startDate ,
offer_endDate , cost_per_person , offer_dst_id};
                tbModel.addRow(tbData);
                con.close();
            }
            catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

        }

    } //GEN-LAST:event_jButton39ActionPerformed

    private void jButton15ComponentResized(java.awt.event.ComponentEvent evt)
{ //GEN-FIRST:event_jButton15ComponentResized
        // TODO add your handling code here:
    } //GEN-LAST:event_jButton15ComponentResized

    private void jButton15ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton15ActionPerformed
        // TODO add your handling code here:
        if( !jTextField20.getText().equals("") ){
            String phNumber = jTextField20.getText();
            String phBrCode = jComboBox3.getSelectedItem().toString();

            String insertPhonesCommand = "INSERT INTO phones VALUES (" +
phBrCode + ", '" + phNumber + "')";

            try {
                Connection con = ConnectionToDb.connectMySqlDB();

                Statement stmt=con.createStatement();
                stmt.executeUpdate(insertPhonesCommand);

                DefaultTableModel tbModel =
(DefaultTableModel)jTable1.getModel();

```

```

        String tbData[] = { phBrCode,phNumber};
        tbModel.addRow(tbData);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

    }
} //GEN-LAST:event_jButton15ActionPerformed

private void jButton9ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton9ActionPerformed
    // TODO add your handling code here:

    if( !jTextField10.getText().equals("") &&
!jTextField11.getText().equals("") && !jTextField12.getText().equals("") &&
!jTextField14.getText().equals("") ){

        String dst_id = jTextField10.getText();
        String dst_name = jTextField11.getText();
        String dst_descr = jTextField12.getText();
        String dst_rtype = jComboBox20.getSelectedItem().toString();
        String dst_language = jTextField14.getText();
        String dst_location = jTextField15.getText();
        String insertDestinationCommand;
        if( !dst_location.equals("") ){
            insertDestinationCommand = "INSERT INTO destination VALUES ("
+ dst_id + "," + dst_name + "," + dst_descr + "," + dst_rtype + "," +
dst_language + "," + dst_location + ");";
        }
        else{
            insertDestinationCommand = "INSERT INTO destination VALUES ("
+ dst_id + "," + dst_name + "," + dst_descr + "," + dst_rtype + "," +
dst_language + "," + NULL );";
        }

        try {
            Connection con = ConnectionToDb.connectMySQLDB();

            Statement stmt=con.createStatement();
            stmt.executeUpdate(insertDestinationCommand);

            DefaultTableModel tbModel =
(DefaultTableModel)jTable3.getModel();
            String tbData[] = new String[6];
            if( !dst_location.equals("") ){
                tbData = new String[] { dst_id, dst_name, dst_descr,
dst_rtype, dst_language, dst_location };
            }
            else{
                tbData = new String[] { dst_id, dst_name, dst_descr,
dst_rtype, dst_language, "NULL" };
            }

            tbModel.addRow(tbData);

```

```

        con.close();
    }
    catch(Exception e){JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

    }

} //GEN-LAST:event_jButton9ActionPerformed

private void jButton24ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton24ActionPerformed
    // TODO add your handling code here:

    // TODO add your handling code here:
    if( !jTextField24.getText().equals("") ){
        String gui_cv = jTextField24.getText();
        String gui_AT = jComboBox7.getSelectedItem().toString();

        String insertGuideCommand = "INSERT INTO guide VALUES ('" +
gui_AT + "','" + gui_cv + "')";

        try {
            Connection con = ConnectionToDb.connectMySqlDB();

            Statement stmt=con.createStatement();
            stmt.executeUpdate(insertGuideCommand);

            DefaultTableModel tbModel =
(DefaultTableModel)jTable5.getModel();
            String tbData[] = { gui_AT , gui_cv };
            tbModel.addRow(tbData);
            con.close();
        }
        catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

    }

} //GEN-LAST:event_jButton24ActionPerformed

private void jButton30ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton30ActionPerformed
    // TODO add your handling code here:

    if( !jTextField25.getText().equals("") ){
        String lng_gui_cv = jTextField25.getText();
        String lng_gui_AT = jComboBox10.getSelectedItem().toString();

        String insertLanguageCommand = "INSERT INTO languages VALUES ('"
+ lng_gui_AT + "','" + lng_gui_cv + "')";

        try {
            Connection con = ConnectionToDb.connectMySqlDB();

            Statement stmt=con.createStatement();
            stmt.executeUpdate(insertLanguageCommand);

```

```

        DefaultTableModel tbModel =
(DefaultTableModel)jTable6.getModel();
        String tbData[] = { lng_gui_AT , lng_gui_cv };
        tbModel.addRow(tbData);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
    }

} //GEN-LAST:event_jButton30ActionPerformed

private void jButton21ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton21ActionPerformed
    // TODO add your handling code here:
    if( !jTextField13.getText().equals("") ){
        String drv_route = jComboBox22.getSelectedItem().toString();
        String drv_experience = jTextField13.getText();
        String drv_AT = jComboBox6.getSelectedItem().toString();
        String drv_license = jComboBox21.getSelectedItem().toString();

        String insertPhonesCommand = "INSERT INTO driver VALUES ('" +
drv_AT + "','" + drv_license + "','" + drv_route + "','" + drv_experience +
");";

        try {
            Connection con = ConnectionToDb.connectMySQLDB();

            Statement stmt=con.createStatement();
            stmt.executeUpdate(insertPhonesCommand);

            DefaultTableModel tbModel =
(DefaultTableModel)jTable7.getModel();
            String tbData[] = { drv_AT , drv_license , drv_route ,
drv_experience };
            tbModel.addRow(tbData);
            con.close();
        }
        catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
    }

} //GEN-LAST:event_jButton21ActionPerformed

private void jButton18ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton18ActionPerformed
    // TODO add your handling code here:

    if( !jTextField21.getText().equals("") ){

        String adm_AT = jComboBox4.getSelectedItem().toString();
        String adm_type = jComboBox5.getSelectedItem().toString();
        String adm_diploma = jTextField21.getText();

        String insertAdminCommand = "INSERT INTO admin VALUES ('" +

```

```

adm_AT + "','" + adm_type + "','" + adm_diploma + "');";

        try {
            Connection con = ConnectionToDb.connectMySqlDB();

            Statement stmt=con.createStatement();
            stmt.executeUpdate(insertAdminCommand);

            DefaultTableModel tbModel =
(DefaultTableModel) jTable8.getModel();
            String tbData[] = { adm_AT , adm_type , adm_diploma };
            tbModel.addRow(tbData);
            con.close();
        }
        catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

    }

} //GEN-LAST:event_jButton18ActionPerformed

private void jButton29ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton29ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable9.getSelectedRow();
    String mng_adm_AT = jTable9.getValueAt(selectedRow, 0).toString();
    String mng_br_code = jTable9.getValueAt(selectedRow, 1).toString();

    String deleteCommand = "delete from manages where mng_adm_AT = '" +
mng_adm_AT + "' AND mng_br_code = '" + mng_br_code + "';";
    int isAdministrative = 0;
    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from admin");

        while(rs.next()){

            String adm_AT = rs.getString("adm_AT");
            String adm_type = rs.getString("adm_type");
            if(adm_AT.equals(mng_adm_AT) &&
adm_type.equals("ADMINISTRATIVE") ){

                isAdministrative = 1;

                JFrame f=new JFrame();
                JOptionPane.showMessageDialog(f,"You cant delete
administrative worker.", "Alert", JOptionPane.WARNING_MESSAGE);
                break;
            }

        }

        if(isAdministrative == 0){
            stmt.executeUpdate(deleteCommand);

```

```

        DefaultTableModel tbModel =
(DefaultTableModel) jTable9.getModel();

        tbModel.removeRow(selectedRow);
    }
    isAdministrative = 0;
    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jButton29ActionPerformed

private void jButton27ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton27ActionPerformed
    // TODO add your handling code here:

    String mng_adm_AT = jComboBox8.getSelectedItem().toString();
    String mng_br_code = jComboBox9.getSelectedItem().toString();

    String insertManagerCommand = "INSERT INTO manages VALUES ('" +
mng_adm_AT + "','" + mng_br_code + "');"

    try {
        Connection con = ConnectionToDb.connectMySQLDB();

        Statement stmt=con.createStatement();
        stmt.executeUpdate(insertManagerCommand);

        DefaultTableModel tbModel =
(DefaultTableModel) jTable9.getModel();
        String tbData[] = { mng_adm_AT , mng_br_code };
        tbModel.addRow(tbData);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jButton27ActionPerformed

private void jButton12ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton12ActionPerformed
    // TODO add your handling code here:

    if( !jTextField17.getText().equals("") &&
!jTextField18.getText().equals("") && !jTextField19.getText().equals("") ){

        String ev_tr_id = jComboBox15.getSelectedItem().toString();
        String ev_start = jTextField17.getText();
        String ev_end = jTextField18.getText();
        String ev_descr = jTextField19.getText();

        String insertEventCommand = "INSERT INTO event VALUES (" +
ev_tr_id + "','" + ev_start + "','" + ev_end + "','" + ev_descr + "');"

        try {

```



```

        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        stmt.executeUpdate(insertEventCommand);

        DefaultTableModel tbModel =
(DefaultTableModel)jTable11.getModel();
        String tbData[] = { ev_tr_id , ev_start , ev_end , ev_descr};
        tbModel.addRow(tbData);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

}

} //GEN-LAST:event_jButton12ActionPerformed

private void jButton35ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton35ActionPerformed
    // TODO add your handling code here:

    if( !jTextField31.getText().equals("") &&
!jTextField32.getText().equals("") && !jTextField33.getText().equals("") ){
        String res_tr_id = jComboBox14.getSelectedItem().toString();
        String res_seatnum = jTextField31.getText();
        String res_name = jTextField32.getText();
        String res_lname = jTextField33.getText();
        String res_isadult = jComboBox19.getSelectedItem().toString();

        String insertReservationCommand = "INSERT INTO reservation VALUES
(" + res_tr_id + "," + res_seatnum + "," + res_name + "," + res_lname +
"," + res_isadult + ")";

        try {
            Connection con = ConnectionToDb.connectMySqlDB();

            Statement stmt=con.createStatement();
            stmt.executeUpdate(insertReservationCommand);

            DefaultTableModel tbModel =
(DefaultTableModel)jTable12.getModel();
            String tbData[] = { res_tr_id , res_seatnum , res_name ,
res_lname , res_isadult};
            tbModel.addRow(tbData);
            con.close();
        }
        catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

    }

} //GEN-LAST:event_jButton35ActionPerformed

private void jButton45ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton45ActionPerformed
    // TODO add your handling code here:

```

```

        if( !jTextField41.getText().equals("") &&
!jTextField42.getText().equals("") && !jTextField43.getText().equals("") &&
!jTextField44.getText().equals("") ){
            String res_offer_id = jTextField20.getText();
            String res_name = jTextField20.getText();
            String res_lname = jTextField20.getText();
            String offer_id_trip = jComboBox3.getSelectedItem().toString();
            String res_deposit = jTextField20.getText();

            String insertReservationOfferCommand = "INSERT INTO
reservation_offers VALUES (" + res_offer_id + "," + res_name + "," + res_lname + "," + offer_id_trip + "," + res_deposit + ");";

            try {
                Connection con = ConnectionToDb.connectMySqlDB();

                Statement stmt=con.createStatement();
                stmt.executeUpdate(insertReservationOfferCommand);

                DefaultTableModel tbModel =
(DefaultTableModel)jTable14.getModel();
                String tbData[] = { res_offer_id + "," + res_name + "," + res_lname + "," + offer_id_trip + "," + res_deposit };
                tbModel.addRow(tbData);
                con.close();
            }
            catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

        }

    } //GEN-LAST:event_jButton45ActionPerformed

    private void jButton11ActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_jButton11ActionPerformed
        // TODO add your handling code here:

        int selectedRow = jTable3.getSelectedRow();
        String dst_id = jTable3.getValueAt(selectedRow, 0).toString();

        String deleteDestinationCommand = "delete from destination where
dst_id= " + dst_id + ";";

        try {
            Connection con = ConnectionToDb.connectMySqlDB();

            Statement stmt=con.createStatement();
            stmt.executeUpdate(deleteDestinationCommand);

            DefaultTableModel tbModel =
(DefaultTableModel)jTable3.getModel();

            tbModel.removeRow(selectedRow);
            con.close();
        }
        catch(Exception e){ JFrame f=new JFrame();

```

```

JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jButton11ActionPerformed

private void jButton23ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton23ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable7.getSelectedRow();
    String drv_AT = jTable7.getValueAt(selectedRow, 0).toString();

    String deleteCommand = "delete from driver where drv_AT= '" + drv_AT
+ "';";

    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        stmt.executeUpdate(deleteCommand);

        DefaultTableModel tbModel =
(DefaultTableModel)jTable7.getModel();

        tbModel.removeRow(selectedRow);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jButton23ActionPerformed

private void jButton14ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton14ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable11.getSelectedRow();
    String ev_tr_id = jTable11.getValueAt(selectedRow, 0).toString();
    String ev_start = jTable11.getValueAt(selectedRow, 1).toString();

    String deleteCommand = "delete from event where ev_tr_id= " +
ev_tr_id + " AND ev_start= '" + ev_start + "';";

    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        stmt.executeUpdate(deleteCommand);

        DefaultTableModel tbModel =
(DefaultTableModel)jTable11.getModel();

        tbModel.removeRow(selectedRow);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

```

```

} //GEN-LAST:event_jButton14ActionPerformed

private void jButton26ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton26ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable5.getSelectedRow();
    String gui_AT = jTable5.getValueAt(selectedRow, 0).toString();

    String deleteCommand = "delete from guide where gui_AT = '" + gui_AT
+ "';";

    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        stmt.executeUpdate(deleteCommand);

        DefaultTableModel tbModel =
(DefaultTableModel)jTable5.getModel();

        tbModel.removeRow(selectedRow);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
} //GEN-LAST:event_jButton26ActionPerformed

private void jButton32ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton32ActionPerformed
    // TODO add your handling code here:
    int selectedRow = jTable6.getSelectedRow();
    String lng_gui_AT = jTable6.getValueAt(selectedRow, 0).toString();
    String lng_language = jTable6.getValueAt(selectedRow, 1).toString();

    String deleteCommand = "delete from languages where lng_gui_AT= '" +
lng_gui_AT + "' AND lng_language= '" + lng_language + "';";

    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        stmt.executeUpdate(deleteCommand);

        DefaultTableModel tbModel =
(DefaultTableModel)jTable6.getModel();

        tbModel.removeRow(selectedRow);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
} //GEN-LAST:event_jButton32ActionPerformed

private void jButton41ActionPerformed(java.awt.event.ActionEvent evt)

```

```

{ //GEN-FIRST:event_jButton41ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable1.getSelectedRow();
    String offer_id = jTable1.getValueAt(selectedRow, 0).toString();

    String deleteCommand = "delete from offers where offer_id= " +
offer_id + ";";

    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        stmt.executeUpdate(deleteCommand);

        DefaultTableModel tbModel =
(DefaultTableModel)jTable13.getModel();

        tbModel.removeRow(selectedRow);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jButton41ActionPerformed

private void jButton17ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton17ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable1.getSelectedRow();
    String ph_br_code = jTable1.getValueAt(selectedRow, 0).toString();
    String ph_number = jTable1.getValueAt(selectedRow, 1).toString();

    String deleteCommand = "delete from phones where ph_br_code= " +
ph_br_code + " AND ph_number='"+ ph_number + "'";

    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        stmt.executeUpdate(deleteCommand);

        DefaultTableModel tbModel =
(DefaultTableModel)jTable1.getModel();

        tbModel.removeRow(selectedRow);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jButton17ActionPerformed

private void jButton38ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton38ActionPerformed

```

```

// TODO add your handling code here:

int selectedRow = jTable12.getSelectedRow();
String res_tr_id = jTable12.getValueAt(selectedRow, 0).toString();
String res_seatnum = jTable12.getValueAt(selectedRow, 1).toString();

String deleteCommand = "delete from reservation where res_tr_id= " +
res_tr_id + " AND res_seatnum="+ res_seatnum +"";

try {
    Connection con = ConnectionToDb.connectMySqlDB();

    Statement stmt=con.createStatement();
    stmt.executeUpdate(deleteCommand);

    DefaultTableModel tbModel =
(DefaultTableModel)jTable12.getModel();

    tbModel.removeRow(selectedRow);
    con.close();
}
catch(Exception e){JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jButton38ActionPerformed

private void jButton47ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton47ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable14.getSelectedRow();
    String res_offer_id = jTable14.getValueAt(selectedRow, 0).toString();

    String deleteCommand = "delete from reservation_offers where
res_offer_id= " + res_offer_id +"";

    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        stmt.executeUpdate(deleteCommand);

        DefaultTableModel tbModel =
(DefaultTableModel)jTable14.getModel();

        tbModel.removeRow(selectedRow);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jButton47ActionPerformed

private void jButton34ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton34ActionPerformed
    // TODO add your handling code here:

```

```

int selectedRow = jTable10.getSelectedRow();
String tr_id = jTable10.getValueAt(selectedRow, 0).toString();

String deleteCommand = "delete from trip where tr_id= " + tr_id +"";

try {
    Connection con = ConnectionToDb.connectMySqlDB();

    Statement stmt=con.createStatement();
    stmt.executeUpdate(deleteCommand);

    DefaultTableModel tbModel =
(DefaultTableModel)jTable10.getModel();

    tbModel.removeRow(selectedRow);
    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
} //GEN-LAST:event_jButton34ActionPerformed

private void jMenuItem18ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem18ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(false);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(true);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);

    try {
        Connection con = ConnectionToDb.connectMySqlDB();
        String showLogProc = "call showLog()";

        CallableStatement stmt = con.prepareCall(showLogProc);
        ResultSet rs = stmt.executeQuery();
        DefaultTableModel tbModel =
(DefaultTableModel)jTable16.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String descr = rs.getString("log_desc");
            String itLname = rs.getString("it_lastname");
            String date = rs.getString("log_date");

            String tbData[] = {descr , itLname , date};

            tbModel.addRow(tbData);
        }

        con.close();
    }
    catch(Exception e){
        System.out.println(e);
    }
}

```

```

    }

    } //GEN-LAST:event_jMenuItem18ActionPerformed

    private void jButton19ActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_jButton19ActionPerformed
        // TODO add your handling code here:

        int selectedRow = jTable8.getSelectedRow();
        if(selectedRow != -1){

            String key = jTable8.getValueAt(selectedRow, 0).toString();

            try {

                Connection con = ConnectionToDb.connectMySqlDB();

                String[] data = new String[ 3 ];

                data[0] = jComboBox4.getSelectedItem().toString();
                data[1] = jComboBox5.getSelectedItem().toString();
                data[2] = jTextField21.getText();

                String command0 = "UPDATE admin set adm_AT='" + data[0] +
                "' where adm_AT= '" + key + "' ";
                String command1 = "UPDATE admin set adm_type= '" +
                data[1] + "' where adm_AT= '" + key + "' ";
                String command2 = "UPDATE admin set adm_diploma= '" +
                data[2] + "' where adm_At= '" + key + "' ";

                if(!data[1].equals("")){
                    CallableStatement stmt = con.prepareCall(command1);
                    stmt.executeUpdate();
                }
                if(!data[2].equals("")){
                    CallableStatement stmt = con.prepareCall(command2);
                    stmt.executeUpdate();
                }
                if(!data[0].equals("")){
                    CallableStatement stmt = con.prepareCall(command0);
                    stmt.executeUpdate();
                }

                //Update table
                CallableStatement stmt = con.prepareCall("select * from
admin");

                ResultSet rs= stmt.executeQuery();
                DefaultTableModel tbModel =
                (DefaultTableModel)jTable8.getModel();
                tbModel.setRowCount(0);
                while(rs.next()){

                    String adm_AT = rs.getString("adm_AT") ;
                    String adm_type = rs.getString("adm_type") ;
                    String adm_diploma = rs.getString("adm_diploma");

```



```

        String tbData[] = {adm_AT, adm_type, adm_diploma };

        tbModel.addRow(tbData);
    }

    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);
}
}

} //GEN-LAST:event_jButton19ActionPerformed

private void jButton10ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton10ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable3.getSelectedRow();
    if(selectedRow != -1){

        String key = jTable3.getValueAt(selectedRow, 0).toString();

        try {

            Connection con = ConnectionToDb.connectMySQLDB();

            String[] data = new String[ 6 ];

            data[0] = jTextField10.getText();
            data[1] = jTextField11.getText();
            data[2] = jTextField12.getText();
            data[3] = jComboBox20.getSelectedItem().toString();
            data[4] = jTextField14.getText();
            data[5] = jTextField15.getText();

            String command0 = "UPDATE destination set dst_id=" +
data[0] + " where dst_id= '" + key + "' ";
            String command1 = "UPDATE destination set dst_name= '" +
data[1] + "' where dst_id= '" + key + "' ";
            String command2 = "UPDATE destination set dst_descr = '"
+ data[2] + "' where dst_id= '" + key + "' ";
            String command3 = "UPDATE destination set dst_rtype = '"
+ data[3] + "' where dst_id= '" + key + "' ";
            String command4 = "UPDATE destination set dst_language =
'" + data[4] + "' where dst_id= '" + key + "' ";
            String command5 = "UPDATE destination set dst_location =
'" + data[5] + "' where dst_id= '" + key + "' ";

            if(!data[1].equals("")){
                CallableStatement stmt = con.prepareCall(command1);
                stmt.executeUpdate();
            }

```

```

        if(!data[2].equals("")){
            CallableStatement stmt = con.prepareStatement(command2);
            stmt.executeUpdate();
        }
        if(!data[3].equals("")){
            CallableStatement stmt = con.prepareStatement(command3);
            stmt.executeUpdate();
        }
        if(!data[4].equals("")){
            CallableStatement stmt = con.prepareStatement(command4);
            stmt.executeUpdate();
        }
        if(!data[5].equals("")){
            CallableStatement stmt = con.prepareStatement(command5);
            stmt.executeUpdate();
        }
        if(!data[0].equals("")){
            CallableStatement stmt = con.prepareStatement(command0);
            stmt.executeUpdate();
        }

        //Update table
        CallableStatement stmt = con.prepareStatement("select * from
destination");

        ResultSet rs= stmt.executeQuery();
        DefaultTableModel tbModel =
(DefaultTableModel) jTable3.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String dst_id = rs.getString("dst_id") ;
            String dst_name = rs.getString("dst_name") ;
            String dst_descr = rs.getString("dst_descr");
            String dst_rtype = rs.getString("dst_rtype");
            String dst_language = rs.getString("dst_language");
            String dst_location = rs.getString("dst_location");

            String tbData[] = {dst_id, dst_name, dst_descr ,
dst_rtype , dst_language , dst_location };

            tbModel.addRow(tbData);
        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
    }

} //GEN-LAST:event_jButton10ActionPerformed

private void jButton22ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton22ActionPerformed
    // TODO add your handling code here:

```

```

int selectedRow = jTable7.getSelectedRow();
if(selectedRow != -1){

    String key = jTable7.getValueAt(selectedRow, 0).toString();

    try {

        Connection con = ConnectionToDb.connectMySQLDB();

        String[] data = new String[ 4 ];

        data[0] = jComboBox6.getSelectedItem().toString();
        data[1] = jComboBox21.getSelectedItem().toString();
        data[2] = jComboBox22.getSelectedItem().toString();
        data[3] = jTextField13.getText();

        String command0 = "UPDATE driver set drv_AT='" + data[0]
+ "' where drv_AT= '" + key + "' ;";
        String command1 = "UPDATE driver set drv_license= '" +
data[1] + "' where drv_AT= '" + key + "' ;";
        String command2 = "UPDATE driver set drv_route = '" +
data[2] + "' where drv_AT= '" + key + "' ;";
        String command3 = "UPDATE driver set drv_experience = " +
data[3] + " where drv_AT= '" + key + "' ;";

        if(!data[1].equals("")){
            CallableStatement stmt = con.prepareCall(command1);
            stmt.executeUpdate();
        }
        if(!data[2].equals("")){
            CallableStatement stmt = con.prepareCall(command2);
            stmt.executeUpdate();
        }
        if(!data[3].equals("")){
            CallableStatement stmt = con.prepareCall(command3);
            stmt.executeUpdate();
        }
        if(!data[0].equals("")){
            CallableStatement stmt = con.prepareCall(command0);
            stmt.executeUpdate();
        }

        //Update table
        CallableStatement stmt = con.prepareCall("select * from
driver");

        ResultSet rs= stmt.executeQuery();
        DefaultTableModel tbModel =
(DefaultTableModel)jTable7.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String drv_AT = rs.getString("drv_AT") ;
            String drv_license = rs.getString("drv_license") ;
            String drv_route = rs.getString("drv_route");
            String drv_experience =

```

```

rs.getString("drv_experience");

        String tbData[] = {drv_AT, drv_license, drv_route ,
drv_experience  };

        tbModel.addRow(tbData);

    }

    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
}
} //GEN-LAST:event_jButton22ActionPerformed

private void jButton13ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton13ActionPerformed
    // TODO add your handling code here:
    int selectedRow = jTable11.getSelectedRow();
    if(selectedRow != -1){

        String key1 = jTable11.getValueAt(selectedRow, 0).toString();
        String key2 = jTable11.getValueAt(selectedRow, 1).toString();

        try {

            Connection con = ConnectionToDb.connectMySQLDB();

            String[] data = new String[ 4 ];

            data[0] = jComboBox15.getSelectedItem().toString();
            data[1] = jTextField17.getText();
            data[2] = jTextField18.getText();
            data[3] = jTextField19.getText();

            //String command0 = "UPDATE event set ev_tr_id=" +
data[0] + " where ev_tr_id= " + key1 + " , ev_start= '"+ key2 +"' ";
            String command1 = "UPDATE event set ev_start= '" +
data[1] + "' where ev_tr_id= " + key1 + " and ev_start= '"+ key2 +"' ";
            String command2 = "UPDATE event set ev_end = '" + data[2]
+ "' where ev_tr_id= " + key1 + " and ev_start= '"+ key2 +"' ";
            String command3 = "UPDATE event set ev_descr = '" +
data[3] + "' where ev_tr_id= " + key1 + " and ev_start= '"+ key2 +"' ";

            if(!data[2].equals("")){
                CallableStatement stmt = con.prepareCall(command2);
                stmt.executeUpdate();
            }
            if(!data[3].equals("")){
                CallableStatement stmt = con.prepareCall(command3);
                stmt.executeUpdate();
            }
            if(!data[1].equals("")){
                CallableStatement stmt = con.prepareCall(command1);
                stmt.executeUpdate();
                key2 = data[1];
            }
        }
    }
}

```

```

        }
        if(!data[0].equals("")){
            String command0 = "UPDATE event set ev_tr_id=" +
data[0] + " where ev_tr_id= " + key1 + " and ev_start= '" + key2 + "' ";
            CallableStatement stmt = con.prepareCall(command0);
            stmt.executeUpdate();

        }

        //Update table
        CallableStatement stmt = con.prepareCall("select * from
event");

        ResultSet rs= stmt.executeQuery();
        DefaultTableModel tbModel =
(DefaultTableModel) jTable11.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String ev_tr_id = rs.getString("ev_tr_id") ;
            String ev_start = rs.getString("ev_start") ;
            String ev_end = rs.getString("ev_end");
            String ev_descr = rs.getString("ev_descr");

            String tbData[] = {ev_tr_id, ev_start, ev_end ,
ev_descr };

            tbModel.addRow(tbData);
        }

        con.close();
    }
    catch(Exception e) { JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
    }
    }//GEN-LAST:event_jButton13ActionPerformed

    private void jButton25ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton25ActionPerformed
    // TODO add your handling code here:
    int selectedRow = jTable5.getSelectedRow();
    if(selectedRow != -1){

        String key = jTable5.getValueAt(selectedRow, 0).toString();

        try {

            Connection con = ConnectionToDb.connectMySQLDB();

            String[] data = new String[ 2 ];

            data[0] = jComboBox7.getSelectedItem().toString();
            data[1] = jTextField24.getText();

            String command0 = "UPDATE guide set gui_AT='" + data[0] +

```

```

" where gui_AT= '" + key + "' ;";
        String command1 = "UPDATE guide set gui_cv= '" + data[1]
+ " where gui_cv= '" + key + "' ;";

        if(!data[1].equals("")){
            CallableStatement stmt = con.prepareCall(command1);
            stmt.executeUpdate();
        }

        if(!data[0].equals("")){
            CallableStatement stmt = con.prepareCall(command0);
            stmt.executeUpdate();
        }

        //Update table
        CallableStatement stmt = con.prepareCall("select * from
guide");

        ResultSet rs= stmt.executeQuery();
        DefaultTableModel tbModel =
(DefaultTableModel) jTable5.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String gui_AT = rs.getString("gui_AT") ;
            String gui_cv = rs.getString("gui_cv") ;

            String tbData[] = {gui_AT, gui_cv };

            tbModel.addRow(tbData);
        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
    }
} //GEN-LAST:event_jButton25ActionPerformed

private void jButton31ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton31ActionPerformed
    // TODO add your handling code here:
    int selectedRow = jTable6.getSelectedRow();
    if(selectedRow != -1){

        String key1 = jTable6.getValueAt(selectedRow, 0).toString();
        String key2 = jTable6.getValueAt(selectedRow, 1).toString();

        try {

            Connection con = ConnectionToDb.connectMySqlDB();

            String[] data = new String[ 2 ];

            data[0] = jComboBox10.getSelectedItem().toString();

```

```

        data[1] = jTextField25.getText();

        String command1 = "UPDATE languages set lng_language= '"
+ data[1] + "' where lng_gui_AT= '" + key1 + "' and lng_language= '" + key2
+ "' ";

        if(!data[1].equals("")){
            CallableStatement stmt = con.prepareCall(command1);
            stmt.executeUpdate();
            key2 = data[1];
        }
        if(!data[0].equals("")){
            String command0 = "UPDATE languages set lng_gui_AT='"
+ data[0] + "' where lng_gui_AT= '" + key1 + "' and lng_language= '" + key2
+ "' ";

            CallableStatement stmt = con.prepareCall(command0);
            stmt.executeUpdate();
        }

        //Update table
        CallableStatement stmt = con.prepareCall("select * from
languages");

        ResultSet rs= stmt.executeQuery();
        DefaultTableModel tbModel =
(DefaultTableModel) jTable6.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String lng_gui_AT = rs.getString("lng_gui_AT") ;
            String lng_language = rs.getString("lng_language") ;

            String tbData[] = {lng_gui_AT, lng_language };

            tbModel.addRow(tbData);
        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);
    }
}
} //GEN-LAST:event_jButton31ActionPerformed

private void jButton28ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton28ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable9.getSelectedRow();
    if(selectedRow != -1){

        String key1 = jTable9.getValueAt(selectedRow, 0).toString();
        String key2 = jTable9.getValueAt(selectedRow, 1).toString();

        try {

```

```

        Connection con = ConnectionToDb.connectMySqlDB();

        String[] data = new String[ 2 ];

        data[0] = jComboBox8.getSelectedItem().toString();
        data[1] = jComboBox9.getSelectedItem().toString();

        String command1 = "UPDATE manages set mng_adm_AT= '" +
data[1] + "' where mng_adm_AT= '" + key1 + "' and mng_br_code= '"+ key2 + " '";

        if(!data[1].equals("")){
            CallableStatement stmt = con.prepareCall(command1);
            stmt.executeUpdate();
            key2 = data[1];
        }
        if(!data[0].equals("")){
            String command0 = "UPDATE manages set mng_adm_AT= '"
+ data[0] + "' where mng_adm_AT= '" + key1 + "' and mng_br_code= '"+ key2 + "
";

            CallableStatement stmt = con.prepareCall(command0);
            stmt.executeUpdate();

        }

        //Update table
        CallableStatement stmt = con.prepareCall("select * from
manages");

        ResultSet rs= stmt.executeQuery();
        DefaultTableModel tbModel =
(DefaultTableModel) jTable9.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String mng_adm_AT = rs.getString("mng_adm_AT") ;
            String mng_br_code = rs.getString("mng_br_code") ;

            String tbData[] = {mng_adm_AT, mng_br_code };

            tbModel.addRow(tbData);

        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

}

} //GEN-LAST:event_jButton28ActionPerformed

private void jButton40ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton40ActionPerformed
    // TODO add your handling code here:
    int selectedRow = jTable13.getSelectedRow();

```



```

        if(selectedRow != -1){

            String key = jTable13.getValueAt(selectedRow, 0).toString();

            try {

                Connection con = ConnectionToDb.connectMySQLDB();

                String[] data = new String[ 5 ];

                data[0] = jTextField16.getText();
                data[1] = jTextField35.getText();
                data[2] = jTextField36.getText();
                data[3] = jTextField37.getText();
                data[4] = jComboBox16.getSelectedItem().toString();

                String command0 = "UPDATE offers set offer_id=" + data[0]
+ " where offer_id= " + key + " ";
                String command1 = "UPDATE offers set offer_startDate= '"
+ data[1] + "' where offer_id= " + key + " ";
                String command2 = "UPDATE offers set offer_endDate = '" +
data[2] + "' where offer_id= " + key + " ";
                String command3 = "UPDATE offers set cost_per_person = "
+ data[3] + " where offer_id= " + key + " ";
                String command4 = "UPDATE offers set offer_dst_id = " +
data[4] + " where offer_id= " + key + " ";

                if(!data[1].equals("")){
                    CallableStatement stmt = con.prepareCall(command1);
                    stmt.executeUpdate();
                }
                if(!data[2].equals("")){
                    CallableStatement stmt = con.prepareCall(command2);
                    stmt.executeUpdate();
                }
                if(!data[3].equals("")){
                    CallableStatement stmt = con.prepareCall(command3);
                    stmt.executeUpdate();
                }
                if(!data[4].equals("")){
                    CallableStatement stmt = con.prepareCall(command4);
                    stmt.executeUpdate();
                }
                if(!data[0].equals("")){
                    CallableStatement stmt = con.prepareCall(command0);
                    stmt.executeUpdate();
                }

                //Update table
                CallableStatement stmt = con.prepareCall("select * from
offers");

                ResultSet rs= stmt.executeQuery();
                DefaultTableModel tbModel =
(DefaultTableModel)jTable13.getModel();

```

```

        tbModel.setRowCount(0);
        while(rs.next()){

            String offer_id = rs.getString("offer_id") ;
            String offer_startDate =
rs.getString("offer_startDate") ;
            String offer_endDate = rs.getString("offer_endDate");
            String cost_per_person =
rs.getString("cost_per_person");
            String offer_dst_id = rs.getString("offer_dst_id");

            String tbData[] = {offer_id, offer_startDate,
offer_endDate , cost_per_person , offer_dst_id };

            tbModel.addRow(tbData);

        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE); }
    }
} //GEN-LAST:event_jButton40ActionPerformed

private void jTable13AncestorAdded(javax.swing.event.AncestorEvent evt)
{//GEN-FIRST:event_jTable13AncestorAdded
    // TODO add your handling code here:
} //GEN-LAST:event_jTable13AncestorAdded

private void jButton16ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton16ActionPerformed
    // TODO add your handling code here:
    int selectedRow = jTable1.getSelectedRow();
    if(selectedRow != -1){

        String key1 = jTable1.getValueAt(selectedRow, 0).toString();
        String key2 = jTable1.getValueAt(selectedRow, 1).toString();

        try {

            Connection con = ConnectionToDb.connectMySqlDB();

            String[] data = new String[ 2 ];

            data[0] = jComboBox3.getSelectedItem().toString();
            data[1] = jTextField20.getText();

            String command1 = "UPDATE phones set ph_number= " +
data[1] + " where ph_br_code= '" + key1 + "' and ph_number= '" + key2 + "' ";

            if(!data[1].equals("")){
                CallableStatement stmt = con.prepareCall(command1);
                stmt.executeUpdate();
                key2 = data[1];
            }

        }
    }
}

```

```

        if(!data[0].equals("")){
            String command0 = "UPDATE phones set ph_br_code= " +
data[0] + " where ph_br_code= '" + key1 + "' and ph_number= '" + key2 + "' ";
            CallableStatement stmt = con.prepareCall(command0);
            stmt.executeUpdate();

        }

        //Update table
        CallableStatement stmt = con.prepareCall("select * from
phones");

        ResultSet rs= stmt.executeQuery();
        DefaultTableModel tbModel =
(DefaultTableModel) jTable1.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String ph_br_code = rs.getString("ph_br_code") ;
            String ph_number = rs.getString("ph_number") ;

            String tbData[] = {ph_br_code, ph_number };

            tbModel.addRow(tbData);

        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
    }
} //GEN-LAST:event_jButton16ActionPerformed

private void jButton37ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton37ActionPerformed
    // TODO add your handling code here:
    int selectedRow = jTable12.getSelectedRow();
    if(selectedRow != -1){

        String key1 = jTable12.getValueAt(selectedRow, 0).toString();
        String key2 = jTable12.getValueAt(selectedRow, 1).toString();

        try {

            Connection con = ConnectionToDb.connectMySQLDB();

            String[] data = new String[ 5 ];

            data[0] = jComboBox14.getSelectedItem().toString();
            data[1] = jTextField31.getText();
            data[2] = jTextField32.getText();
            data[3] = jTextField33.getText();
            data[4] = jComboBox19.getSelectedItem().toString();

            String command1 = "UPDATE reservation set res_seatnum= "
+ data[1] + " where res_tr_id= " + key1 + " and res_seatnum= " + key2 + " ";
            String command2 = "UPDATE reservation set res_name= '" +
data[1] + "' where res_tr_id= " + key1 + " and res_seatnum= " + key2 + " ";

```

```

        String command3 = "UPDATE reservation set res_lname= '" +
data[1] + "' where res_tr_id= " + key1 + " and res_seatnum= " + key2 + " ";
        String command4 = "UPDATE reservation set res_isadult= '"
+ data[1] + "' where res_tr_id= " + key1 + " and res_seatnum= " + key2 + " ";

        if(!data[1].equals("")){
            CallableStatement stmt = con.prepareCall(command1);
            stmt.executeUpdate();
            key2 = data[1];
        }
        if(!data[2].equals("")){
            CallableStatement stmt = con.prepareCall(command2);
            stmt.executeUpdate();
            key2 = data[1];
        }
        if(!data[3].equals("")){
            CallableStatement stmt = con.prepareCall(command3);
            stmt.executeUpdate();
            key2 = data[1];
        }
        if(!data[4].equals("")){
            CallableStatement stmt = con.prepareCall(command4);
            stmt.executeUpdate();
            key2 = data[1];
        }
        if(!data[0].equals("")){
            String command0 = "UPDATE reservation set res_tr_id="
+ data[0] + " where res_tr_id= '" + key1 + "' and res_seatnum= " + key2 + " ";
            CallableStatement stmt = con.prepareCall(command0);
            stmt.executeUpdate();
        }

        //Update table
        CallableStatement stmt = con.prepareCall("select * from
reservation");

        ResultSet rs= stmt.executeQuery();
        DefaultTableModel tbModel =
(DefaultTableModel) jTable12.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String res_tr_id = rs.getString("res_tr_id") ;
            String res_seatnum = rs.getString("res_seatnum") ;
            String res_name = rs.getString("res_name") ;
            String res_lname = rs.getString("res_lname") ;
            String res_isadult = rs.getString("res_isadult") ;

            String tbData[] = {res_tr_id, res_seatnum ,res_name
, res_lname , res_isadult};

            tbModel.addRow(tbData);
        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();

```

```

JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE); }
    }
} //GEN-LAST:event_jButton37ActionPerformed

private void jButton46ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton46ActionPerformed
    // TODO add your handling code here:
    int selectedRow = jTable14.getSelectedRow();
    if(selectedRow != -1){

        String key1 = jTable14.getValueAt(selectedRow, 0).toString();

        try {

            Connection con = ConnectionToDb.connectMySqlDB();

            String[] data = new String[ 5 ];

            data[0] = jTextField41.getText();
            data[1] = jTextField42.getText();
            data[2] = jTextField43.getText();
            data[3] = jComboBox18.getSelectedItem().toString();
            data[4] = jTextField44.getText();

            String command0 = "UPDATE reservation_offers set
res_offer_id = " + data[0] + " where res_offer_id = " + key1 + " ";
            String command1 = "UPDATE reservation_offers set res_name
= '" + data[1] + "' where res_offer_id = " + key1 + " ";
            String command2 = "UPDATE reservation_offers set
res_lname = '" + data[2] + "' where res_offer_id = " + key1 + " ";
            String command3 = "UPDATE reservation_offers set
offer_id_trip = " + data[3] + " where res_offer_id = " + key1 + " ";
            String command4 = "UPDATE reservation_offers set
res_deposit = " + data[4] + " where res_offer_id = " + key1 + " ";

            if(!data[1].equals("")){
                CallableStatement stmt = con.prepareCall(command1);
                stmt.executeUpdate();
            }
            if(!data[2].equals("")){
                CallableStatement stmt = con.prepareCall(command2);
                stmt.executeUpdate();
            }
            if(!data[3].equals("")){
                CallableStatement stmt = con.prepareCall(command3);
                stmt.executeUpdate();
            }
            if(!data[4].equals("")){
                CallableStatement stmt = con.prepareCall(command4);
                stmt.executeUpdate();
            }
            if(!data[0].equals("")){

```

```

        CallableStatement stmt = con.prepareCall(command0);
        stmt.executeUpdate();

    }

    //Update table
    CallableStatement stmt = con.prepareCall("select * from
reservation_offers");
    ResultSet rs= stmt.executeQuery();
    DefaultTableModel tbModel =
(DefaultTableModel)jTable14.getModel();
    tbModel.setRowCount(0);
    while(rs.next()){

        String res_offer_id = rs.getString("res_offer_id") ;
        String res_name = rs.getString("res_name") ;
        String res_lname = rs.getString("res_lname") ;
        String offer_id_trip = rs.getString("offer_id_trip")
;

        String res_deposit = rs.getString("res_deposit") ;

        String tbData[] = {res_offer_id, res_name ,res_lname
,offer_id_trip , res_deposit};

        tbModel.addRow(tbData);
    }

    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE); }
}
} //GEN-LAST:event_jButton46ActionPerformed

private void jButton33ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton33ActionPerformed
    // TODO add your handling code here:
    int selectedRow = jTable10.getSelectedRow();
    if(selectedRow != -1){

        String key = jTable10.getValueAt(selectedRow, 0).toString();

        try {

            Connection con = ConnectionToDb.connectMySQLDB();

            String[] data = new String[ 8 ];

            data[0] = jTextField26.getText();
            data[1] = jTextField27.getText();
            data[2] = jTextField28.getText();
            data[3] = jTextField29.getText();
            data[4] = jTextField30.getText();
            data[5] = jComboBox11.getSelectedItem().toString();
            data[6] = jComboBox12.getSelectedItem().toString();
            data[7] = jComboBox13.getSelectedItem().toString();

```

```

        String command0 = "UPDATE trip set tr_id= " + data[0] + "
where tr_id= " + key + " ";
        String command1 = "UPDATE trip set tr_departure= '" +
data[1] + "' where tr_id= " + key + " ";
        String command2 = "UPDATE trip set tr_return= '" +
data[2] + "' where tr_id= " + key + " ";
        String command3 = "UPDATE trip set tr_maxseats= " +
data[3] + " where tr_id= " + key + " ";
        String command4 = "UPDATE trip set tr_cost= " + data[4] +
" where tr_id= " + key + " ";
        String command5 = "UPDATE trip set tr_br_code= " +
data[5] + " where tr_id= " + key + " ";
        String command6 = "UPDATE trip set tr_gui_AT= '" +
data[6] + "' where tr_id= " + key + " ";
        String command7 = "UPDATE trip set tr_drv_AT= '" +
data[7] + "' where tr_id= '" + key + "' ";

        if(!data[1].equals("")){
            CallableStatement stmt = con.prepareCall(command1);
            stmt.executeUpdate();
        }
        if(!data[2].equals("")){
            CallableStatement stmt = con.prepareCall(command2);
            stmt.executeUpdate();
        }
        if(!data[3].equals("")){
            CallableStatement stmt = con.prepareCall(command3);
            stmt.executeUpdate();
        }
        if(!data[4].equals("")){
            CallableStatement stmt = con.prepareCall(command4);
            stmt.executeUpdate();
        }
        if(!data[5].equals("")){
            CallableStatement stmt = con.prepareCall(command5);
            stmt.executeUpdate();
        }
        if(!data[6].equals("")){
            CallableStatement stmt = con.prepareCall(command6);
            stmt.executeUpdate();
        }
        if(!data[7].equals("")){
            CallableStatement stmt = con.prepareCall(command7);
            stmt.executeUpdate();
        }
        if(!data[0].equals("")){
            CallableStatement stmt = con.prepareCall(command0);
            stmt.executeUpdate();
        }

        //Update table
        CallableStatement stmt = con.prepareCall("select * from
trip");

```

```

        ResultSet rs= stmt.executeQuery();
        DefaultTableModel tbModel =
(DefaultTableModel) jTable10.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String tr_id = rs.getString("tr_id") ;
            String tr_departure = rs.getString("tr_departure") ;
            String tr_return = rs.getString("tr_return");
            String tr_maxseats = rs.getString("tr_maxseats");
            String tr_cost = rs.getString("tr_drv_AT") ;
            String tr_br_code = rs.getString("tr_drv_AT") ;
            String tr_gui_AT = rs.getString("tr_drv_AT");
            String tr_drv_AT = rs.getString("tr_drv_AT");

            String tbData[] = {tr_id, tr_departure, tr_return ,
tr_maxseats, tr_drv_AT, tr_drv_AT, tr_drv_AT , tr_drv_AT };

            tbModel.addRow(tbData);

        }

        con.close();
    }
    catch(SQLException e){
        JFrame f=new JFrame();

JOptionPane.showMessageDialog(f,e.getMessage(),"Alert",JOptionPane.WARNING_ME
SSAGE);
    }

}

} //GEN-LAST:event_jButton33ActionPerformed

private void jTextField26ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jTextField26ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jTextField26ActionPerformed

private void jTextField22ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jTextField22ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jTextField22ActionPerformed

private void jMenuItem19ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jMenuItem19ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(true);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    dataMenuPanel.removeAll();

```



```

dataMenuPanel.add(travelToPanel).setVisible(true);

jComboBox23.removeAllItems();
jComboBox24.removeAllItems();

try {
    Connection con = ConnectionToDb.connectMySqlDB();

    Statement stmt=con.createStatement();
    ResultSet rs=stmt.executeQuery("select * from travelTo");
    DefaultTableModel tbModel =
(DefaultTableModel)jTable23.getModel();
    tbModel.setRowCount(0);
    while(rs.next()){
        String to_tr_id = String.valueOf(rs.getInt("to_tr_id"));
        String to_dst_id = String.valueOf(rs.getInt("to_dst_id"));

        String to_arrival = rs.getDate("to_arrival").toString();
        String to_departure =
rs.getDate("to_departure").toString();

        String tbData[] = {to_tr_id, to_dst_id ,to_arrival ,
to_departure };

        tbModel.addRow(tbData);
    }

    rs=stmt.executeQuery("select tr_id from trip");
    while(rs.next()){

        //String br_code = String.valueOf(rs.getInt("br_code"));
        String tr_id = rs.getString("tr_id");
        jComboBox23.addItem(tr_id);

    }

    rs=stmt.executeQuery("select dst_id from destination");
    while(rs.next()){

        //String br_code = String.valueOf(rs.getInt("br_code"));
        String dst_id = rs.getString("dst_id");
        jComboBox24.addItem(dst_id);

    }

    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jMenuItem19ActionPerformed

private void jButton48ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton48ActionPerformed
    // TODO add your handling code here:
    if( !jTextField22.getText().equals("") &&
!jTextField23.getText().equals("") ){

```

```

String to_tr_id = jComboBox23.getSelectedItem().toString();
String to_dst_id = jComboBox24.getSelectedItem().toString();
String to_arrival = jTextField22.getText();
String to_departure = jTextField23.getText();

String insertTravelToCommand = "INSERT INTO travelTo VALUES (" +
to_tr_id + " , " + to_dst_id + " , '" + to_arrival + "' , '" + to_departure
+"')";

try {
    Connection con = ConnectionToDb.connectMySQLDB();

    Statement stmt=con.createStatement();
    stmt.executeUpdate(insertTravelToCommand);

    DefaultTableModel tbModel =
(DefaultTableModel)jTable23.getModel();
    String tbData[] = { to_tr_id , to_dst_id , to_arrival ,
to_departure };
    tbModel.addRow(tbData);
    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
}
} //GEN-LAST:event_jButton48ActionPerformed

private void jButton49ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton49ActionPerformed
    // TODO add your handling code here:
    int selectedRow = jTable23.getSelectedRow();
    if(selectedRow != -1){

        String key1 = jTable23.getValueAt(selectedRow, 0).toString();
        String key2 = jTable23.getValueAt(selectedRow, 1).toString();

        try {

            Connection con = ConnectionToDb.connectMySQLDB();

            String[] data = new String[ 4 ];

            data[0] = jComboBox23.getSelectedItem().toString();
            data[1] = jComboBox24.getSelectedItem().toString();
            data[2] = jTextField22.getText();
            data[3] = jTextField23.getText();

            String command1 = "UPDATE travelTo set to_dst_id= " +
data[1] + " where to_tr_id = " + key1 + " and to_dst_id = " + key2 + " ";
            String command2 = "UPDATE travelTo set to_arrival= '" +
data[2] + "' where to_tr_id = " + key1 + " and to_dst_id = " + key2 + " ";
            String command3 = "UPDATE travelTo set to_departure= '" +
data[3] + "' where to_tr_id = " + key1 + " and to_dst_id = " + key2 + " ";

```

```

        if(!data[2].equals("")){
            CallableStatement stmt = con.prepareCall(command2);
            stmt.executeUpdate();
        }
        if(!data[3].equals("")){
            CallableStatement stmt = con.prepareCall(command3);
            stmt.executeUpdate();
        }
        if(!data[1].equals("")){
            CallableStatement stmt = con.prepareCall(command1);
            stmt.executeUpdate();
            key2 = data[1];
        }
        if(!data[0].equals("")){
            String command0 = "UPDATE travelTo set to_tr_id= " +
data[0] + " where to_tr_id = " + key1 + " and to_dst_id = " + key2 + " ";
            CallableStatement stmt = con.prepareCall(command0);
            stmt.executeUpdate();
        }

        //Update table
        CallableStatement stmt = con.prepareCall("select * from
travelTo");

        ResultSet rs= stmt.executeQuery();
        DefaultTableModel tbModel =
(DefaultTableModel) jTable23.getModel();
        tbModel.setRowCount(0);
        while(rs.next()){

            String to_tr_id = rs.getString("to_tr_id") ;
            String to_dst_id = rs.getString("to_dst_id") ;
            String to_arrival = rs.getString("to_arrival");
            String to_departure = rs.getString("to_departure");

            String tbData[] = {to_tr_id, to_dst_id, to_arrival ,
to_departure};

            tbModel.addRow(tbData);
        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
    }
} //GEN-LAST:event_jButton49ActionPerformed

private void jButton50ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton50ActionPerformed
    // TODO add your handling code here:
    int selectedRow = jTable23.getSelectedRow();
    String to_tr_id = jTable23.getValueAt(selectedRow, 0).toString();

```

```

        String to_dst_id = jTable23.getValueAt(selectedRow, 1).toString();

        String deleteCommand = "delete from travelTo where to_tr_id= " +
to_tr_id + " AND to_dst_id= "+ to_dst_id + " ";

        try {
            Connection con = ConnectionToDb.connectMySqlDB();

            Statement stmt=con.createStatement();
            stmt.executeUpdate(deleteCommand);

            DefaultTableModel tbModel =
(DefaultTableModel)jTable23.getModel();

            tbModel.removeRow(selectedRow);
            con.close();
        }
        catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

    }//GEN-LAST:event_jButton50ActionPerformed

    private void jButton20ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton20ActionPerformed
        // TODO add your handling code here:

        int selectedRow = jTable8.getSelectedRow();
        String adm_AT = jTable8.getValueAt(selectedRow, 0).toString();
        String selectWorker = "select wrk_name, wrk_lame from worker where
wrk_AT = '" + adm_AT + "'";

        try{
            Connection con = ConnectionToDb.connectMySqlDB();
            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery(selectWorker);

            while(rs.next()){
                String fName = rs.getString("wrk_name");
                String lName = rs.getString("wrk_lame");
                String deleteAdminProc = "call delete_admin('" + fName + "'" + ",
'" + lName + "'" + ")";
                CallableStatement stmt2 = con.prepareCall(deleteAdminProc);
                ResultSet rs2 = stmt2.executeQuery();
            }

            con.close();
        }catch(SQLException e){
            JFrame f=new JFrame();

JOptionPane.showMessageDialog(f,e.getMessage(),"Alert",JOptionPane.WARNING_ME
SSAGE);
        }

    }//GEN-LAST:event_jButton20ActionPerformed

    private void workersPanelAncestorAdded(javax.swing.event.AncestorEvent

```

```

evt) { //GEN-FIRST:event_workersPanelAncestorAdded
    // TODO add your handling code here:
} //GEN-LAST:event_workersPanelAncestorAdded

private void jButton8ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton8ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable4.getSelectedRow();
    String wrk_AT = jTable4.getValueAt(selectedRow, 0).toString();
    String wrk_br_code = jTable4.getValueAt(selectedRow, 4).toString();

    String deleteCommand = "delete from worker where wrk_AT = '" + wrk_AT
+ "' AND wrk_br_code = '" + wrk_br_code + "';";
    int isAdministrative = 0;
    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from admin");

        while(rs.next()){

            String adm_AT = rs.getString("adm_AT");
            String adm_type = rs.getString("adm_type");
            if(adm_AT.equals(wrk_AT) && adm_type.equals("ADMINISTRATIVE"))
        ){

            isAdministrative = 1;

            JFrame f=new JFrame();
            JOptionPane.showMessageDialog(f,"You cant delete
administrative worker.", "Alert", JOptionPane.WARNING_MESSAGE);
            break;

        }

    }

    if(isAdministrative == 0){
        stmt.executeUpdate(deleteCommand);
        DefaultTableModel tbModel =
(DefaultTableModel)jTable4.getModel();

        tbModel.removeRow(selectedRow);
    }
    isAdministrative = 0;
    con.close();
}

catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

} //GEN-LAST:event_jButton8ActionPerformed
private void jMenuItem20ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem20ActionPerformed
    // TODO add your handling code here:
    // Admin Card
    dataMenuPanel.setVisible(false);
}

```

```

branchesMenuPanel.setVisible(false);
offersMenuPanel.setVisible(false);
logsMenuPanel.setVisible(false);
adminMenuPanel.setVisible(true);
driverMenuPanel.setVisible(false);
GuideMenuPanel.setVisible(false);

jComboBox25.removeAllItems();

try{
    Connection con = ConnectionToDb.connectMySQLDB();
    Statement stmt = con.createStatement();
    ResultSet rs = stmt.executeQuery("select distinct adm_AT from
admin");

    while(rs.next()){
        String admin_at = rs.getString("adm_AT");
        jComboBox25.addItem(admin_at);
    }

    con.close();

}catch(Exception e){System.out.println(e);}

//add default label names

} //GEN-LAST:event_jMenuItem20ActionPerformed

private void jMenuItem4ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jMenuItem4ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jMenuItem4ActionPerformed

private void jComboBox25ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jComboBox25ActionPerformed
    // TODO add your handling code here:

    if(jComboBox25.getSelectedItem() != null){
        String selectedItem = jComboBox25.getSelectedItem().toString();

        String adminCardProcedure = "call admin_card('" + selectedItem +
        "')";

        try{
            Connection con = ConnectionToDb.connectMySQLDB();
            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery(adminCardProcedure);

            while(rs.next()){
                jLabel181.setText(rs.getString("First Name"));
                jLabel182.setText(rs.getString("Last Name"));
                jLabel183.setText(rs.getString("AT"));
            }
        }
    }
}

```

```

        jLabel184.setText(rs.getString("Branch Code"));
        jLabel185.setText(rs.getString("Salary"));
        jLabel186.setText(rs.getString("Type"));
        jTextArea1.setText(rs.getString("Diploma"));
    }

    con.close();
} catch (Exception e) {System.out.println(e);}
}

} //GEN-LAST:event_jComboBox25ActionPerformed

private void jComboBox4ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jComboBox4ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jComboBox4ActionPerformed

private void jMenuItem21ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem21ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(false);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(true);
    GuideMenuPanel.setVisible(false);

    jComboBox26.removeAllItems();

    try{
        Connection con = ConnectionToDb.connectMySqlDB();
        Statement stmt=con.createStatement();
        ResultSet rs = stmt.executeQuery("select distinct drv_AT from
driver");
        while(rs.next()){
            String driver_id = rs.getString("drv_AT");
            jComboBox26.addItem(driver_id);
        }
        con.close();

    } catch (Exception e) {System.out.println(e);}

} //GEN-LAST:event_jMenuItem21ActionPerformed

private void jMenuItem22ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem22ActionPerformed
    // TODO add your handling code here:
    // TODO add your handling code here:
    dataMenuPanel.setVisible(false);
    branchesMenuPanel.setVisible(false);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(true);

```

```

jComboBox27.removeAllItems();

try{
    Connection con = ConnectionToDb.connectMySqlDB();
    Statement stmt=con.createStatement();
    ResultSet rs = stmt.executeQuery("select distinct gui_AT from
guide");
    while(rs.next()){
        String guide_id = rs.getString("gui_AT");
        jComboBox27.addItem(guide_id);
    }
    con.close();

}catch(Exception e){System.out.println(e);}

} //GEN-LAST:event_jMenuItem22ActionPerformed

private void jComboBox26ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jComboBox26ActionPerformed
    // TODO add your handling code here:
    if(jComboBox26.getSelectedItem() != null){

        String selectedItem = jComboBox26.getSelectedItem().toString();
        System.out.println(selectedItem);
        String driverCard = "call driver_card('" + selectedItem + "')";
        //String sl = "select wrk_name from worker where
wrk_AT="+selectedItem;
        System.out.println(driverCard);

        try{
            Connection con = ConnectionToDb.connectMySqlDB();
            CallableStatement stmt = con.prepareCall(driverCard);
            boolean results = stmt.execute();

            ResultSet rs = stmt.getResultSet();

            DefaultTableModel tbModelData =
(DefaultTableModel)jTable24.getModel();
            tbModelData.setRowCount(0);

            //Result Set #1
            while(rs.next()){
                jLabel199.setText(rs.getString("First Name"));
                jLabel1100.setText(rs.getString("Last Name"));
                jLabel1101.setText(rs.getString("AT"));
                jLabel1102.setText(rs.getString("Branch Code"));
                jLabel1103.setText(rs.getString("Salary"));
                jLabel1104.setText(rs.getString("License"));
                jLabel1105.setText(rs.getString("Route"));
                jLabel1106.setText(rs.getString("Experience"));
            }

            //Result Set #2

```



```

        results = stmt.getMoreResults();

        if(results == true){
            rs = stmt.getResultSet();
            while(rs.next()){
                String tripId = rs.getString("Trip Id");
                String startDate = rs.getString("departure");
                String returnDate = rs.getString("Return");

                String tbData[] = {tripId, startDate, returnDate};
                tbModelData.addRow(tbData);
            }
        }
        con.close();

    }catch(Exception e){System.out.println(e);}
}

} //GEN-LAST:event_jComboBox26ActionPerformed

private void jComboBox27ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jComboBox27ActionPerformed
    // TODO add your handling code here:
    if(jComboBox27.getSelectedItem() != null){

        String selectedItem = jComboBox27.getSelectedItem().toString();
        String guideCard = "call guide_card('" + selectedItem + "')";
        //String sl = "select wrk_name from worker where
wrk_AT="+selectedItem;

        try{
            Connection con = ConnectionToDb.connectMySqlDB();
            CallableStatement stmt = con.prepareCall(guideCard);
            boolean results = stmt.execute();

            ResultSet rs = stmt.getResultSet();

            DefaultTableModel tbModelDataLang =
(DefaultTableModel)jTable25.getModel();
            tbModelDataLang.setRowCount(0);

            DefaultTableModel tbModelDataTrips =
(DefaultTableModel)jTable26.getModel();
            tbModelDataTrips.setRowCount(0);

            //Result Set #1
            while(rs.next()){
                jLabel114.setText(rs.getString("First Name"));
                jLabel115.setText(rs.getString("Last Name"));
                jLabel116.setText(rs.getString("AT"));
                jLabel117.setText(rs.getString("Branch Code"));
                jLabel118.setText(rs.getString("Salary"));
            }

            //Result Set #2

```

```

        results = stmt.getMoreResults();

        if(results == true){
            rs = stmt.getResultSet();
            while(rs.next()){
                String language = rs.getString("lng_language");

                String tbData[] = {language};
                tbModelDataLang.addRow(tbData);
            }
        }

        //Result Set #3
        results = stmt.getMoreResults();

        if(results == true){
            rs = stmt.getResultSet();
            while(rs.next()){
                String tripId = rs.getString("Trip Id");
                String startDate = rs.getString("departure");
                String returnDate = rs.getString("Return");

                String tbData[] = {tripId, startDate, returnDate};
                tbModelDataTrips.addRow(tbData);
            }
        }

        //Result Set #4
        results = stmt.getMoreResults();

        if(results == true){
            rs = stmt.getResultSet();
            while(rs.next()){
                jTextArea2.setText(rs.getString("CV"));
            }
        }

        con.close();

    }catch(Exception e){System.out.println(e);}
}
} //GEN-LAST:event_jComboBox27ActionPerformed

private void jMenuItem23ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem23ActionPerformed
    // TODO add your handling code here:
    dataMenuPanel.setVisible(false);
    branchesMenuPanel.setVisible(true);
    offersMenuPanel.setVisible(false);
    logsMenuPanel.setVisible(false);
    adminMenuPanel.setVisible(false);
    driverMenuPanel.setVisible(false);
    GuideMenuPanel.setVisible(false);
    branchesMenuPanel.removeAll();
    branchesMenuPanel.add(availableTripsPanel).setVisible(true);
}
} //GEN-LAST:event_jMenuItem23ActionPerformed

```

```

jComboBox28.removeAllItems();
try {
    Connection con = ConnectionToDb.connectMySqlDB();

    /*Branch*/
    Statement stmt=con.createStatement();
    ResultSet rs=stmt.executeQuery("select br_code from branch");

    //Remove all items from combobox

    while(rs.next()){

        String br_code = rs.getString("br_code");
        jComboBox28.addItem(br_code);

    }
    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
} //GEN-LAST:event_jMenuItem23ActionPerformed

private void jComboBox28MouseClicked(java.awt.event.MouseEvent evt)
{ //GEN-FIRST:event_jComboBox28MouseClicked
    // TODO add your handling code here:
} //GEN-LAST:event_jComboBox28MouseClicked

private void jComboBox28ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jComboBox28ActionPerformed
    // TODO add your handling code here:
    //For availabletripsBranch

} //GEN-LAST:event_jComboBox28ActionPerformed

private void
startDateTextField1ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-
FIRST:event_startDateTextField1ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_startDateTextField1ActionPerformed

private void endDateTextField1ActionPerformed(java.awt.event.ActionEvent
evt) { //GEN-FIRST:event_endDateTextField1ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_endDateTextField1ActionPerformed

private void jButton51ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton51ActionPerformed
    // TODO add your handling code here:
    String startDate = startDateTextField1.getText();
    String endDate = endDateTextField1.getText();
    String brCode = jComboBox28.getSelectedItem().toString();

    String procedureShowAvailableTrips = "call showAvailableTrips(" +
brCode + "," + " \" " + startDate + " \" " + "," + " \" " + endDate + " \"
" + ")";

```

```

        try {
            Connection con = ConnectionToDb.connectMySQLDB();

            DefaultTableModel tbModelData =
(DefaultTableModel)jTable27.getModel();
            tbModelData.setRowCount(0);

            CallableStatement stmt =
con.prepareCall(procedureShowAvailableTrips);
            boolean results = stmt.execute();
            while (results) {
                ResultSet rs = stmt.getResultSet();
                // process result set

                while(rs.next()){
                    String tripCost = rs.getString("tripCost");
                    String maxSeats = rs.getString("maxSeats");
                    String reservCount = rs.getString("reservCount");
                    String emptySeats = rs.getString("emptySeats");
                    String drivLname = rs.getString("drivLname");
                    String drivName = rs.getString("drivName");
                    String GuideLname = rs.getString("GuideLname");
                    String GuideName = rs.getString("GuideName");
                    String departureDate = rs.getString("departureDate");
                    String returnDate = rs.getString("returnDate");

                    String tbData[] = {tripCost , maxSeats , reservCount
,emptySeats ,drivLname ,drivName ,GuideLname , GuideName, departureDate,
returnDate};

                    tbModelData.addRow(tbData);
                }

                results = stmt.getMoreResults();
            }

            con.close();
        }
        catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

    }//GEN-LAST:event_jButton51ActionPerformed

    private void jComboBox1ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jComboBox1ActionPerformed
        // TODO add your handling code here:
        // TODO add your handling code here:

        //Because of combobox reset the value is null and we have an
exception
        //Use if statement to check when the combobox is not empty
        if(jComboBox1.getSelectedItem() != null){

            String selectedItem = jComboBox1.getSelectedItem().toString();

```

```

String workersInfoProc = "call branchInfo(" + selectedItem + ")";
String branchStatsProc = "call branch_stats(" + selectedItem +
");";

String branchDetails = "call showBranchDetails(" + selectedItem +
");";

try {
    Connection con = ConnectionToDb.connectMySQLDB();

    CallableStatement stmt = con.prepareCall(workersInfoProc);
    boolean results = stmt.execute();

    ResultSet rs = stmt.getResultSet();

    DefaultTableModel tbModelWorkerInfo =
(DefaultTableModel) jTable17.getModel();
    tbModelWorkerInfo.setRowCount(0);

    DefaultTableModel tbBrDetails =
(DefaultTableModel) jTable18.getModel();
    tbBrDetails.setRowCount(0);

    DefaultTableModel tbModelSalary =
(DefaultTableModel) jTable19.getModel();
    tbModelSalary.setRowCount(0);

    DefaultTableModel tbModelFinancial =
(DefaultTableModel) jTable20.getModel();
    tbModelFinancial.setRowCount(0);

    //Worker Data
    while(rs.next()){
        String firstName = rs.getString("First Name");
        String lastName = rs.getString("Last Name");
        String salary = String.valueOf(rs.getDouble("Salary") );

        String tbData[] = {firstName, lastName, salary};

        tbModelWorkerInfo.addRow(tbData);
    }

    //Total Salary
    results = stmt.getMoreResults();

    if(results == true){
        rs = stmt.getResultSet();
        while(rs.next()){
            String total = rs.getString("SUM(wrk_salary)");

            String tbData[] = {total};
            tbModelSalary.addRow(tbData);
        }
    }

    //Branch details

```

```

        Statement stmt2 = con.createStatement();
        ResultSet rs2 = stmt2.executeQuery(branchDetails);

        while(rs2.next()){
            String code = String.valueOf(rs2.getInt("Code") );
            String number = String.valueOf(rs2.getInt("Number") );
            String street = rs2.getString("Street");
            String city = rs2.getString("City");
            String admin = rs2.getString("admin (FN)");
            String lastName = rs2.getString("Last name (LN)");
            String numOfRes = rs2.getString("Num of reservations");
            String income = rs2.getString("Income");

            String tbDetails[] = {code, number, street, city, admin,
lastName, numOfRes, income};

            tbBrDetails.addRow(tbDetails);
        }

        Statement stmt3 = con.createStatement();
        ResultSet rs3 = stmt3.executeQuery(branchStatsProc);
        while(rs3.next()){
            String income = String.valueOf(rs3.getInt("Income"));
            String total_salaries =
String.valueOf(rs3.getInt("expenses"));
            String turnover = String.valueOf(rs3.getInt("Turnover"));

            String tbStats[] = {income, total_salaries, turnover};
            tbModelFinancial.addRow(tbStats);

        }

        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();

JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

    }

    //GEN-LAST:event_jComboBox1ActionPerformed

    private void jComboBox1MouseClicked(java.awt.event.MouseEvent evt)
{//GEN-FIRST:event_jComboBox1MouseClicked

    //GEN-LAST:event_jComboBox1MouseClicked

    private void jMenuItem6MousePressed(java.awt.event.MouseEvent evt) {//GEN-
FIRST:event_jMenuItem6MousePressed
        // TODO add your handling code here:
        JFrame logoutFrame = new JFrame();
        int choice = JOptionPane.showConfirmDialog(logoutFrame, "Are you sure
you want to logout?", "Confirmation", JOptionPane.YES_NO_OPTION);
        if (choice == JOptionPane.YES_OPTION) {
            try{

```

```

        Connection con = ConnectionToDb.connectMySQLDB();

        //Delete temporary
        Statement stmt3=con.createStatement();
        stmt3.execute("truncate table templogin;");
        dispose();
        new Login().setVisible(true);
    }catch(Exception e){System.out.println(e);}
}else
    logoutFrame.dispose();

} //GEN-LAST:event_jMenu6MousePressed

private void formWindowClosing(java.awt.event.WindowEvent evt) { //GEN-FIRST:event_formWindowClosing
    // TODO add your handling code here:
    JFrame exitFrame = new JFrame();
    int choice = JOptionPane.showConfirmDialog(exitFrame,
        "Are you sure you want to close this window?", "Close Window?",
        JOptionPane.YES_NO_OPTION,
        JOptionPane.QUESTION_MESSAGE);
    if (choice == JOptionPane.YES_OPTION){
        try{
            Connection con = ConnectionToDb.connectMySQLDB();

            //Delete temporary
            Statement stmt=con.createStatement();
            stmt.execute("truncate table templogin;");
            System.exit(0);

        }catch(Exception e){System.out.println(e);}
        System.exit(0);
    }else
        exitFrame.dispose();
} //GEN-LAST:event_formWindowClosing

private void jTextField47ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jTextField47ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jTextField47ActionPerformed

private void jTextField45ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jTextField45ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jTextField45ActionPerformed

private void jTextField34ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jTextField34ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jTextField34ActionPerformed

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton3ActionPerformed
    // TODO add your handling code here:
    String at = jTextField34.getText();
    String fn = jTextField45.getText();
    String ln = jTextField46.getText();

```

```

        String sal = jTextField47.getText();
        String bCode = jComboBox29.getSelectedItem().toString();
        String pass = jTextField49.getText();
        String sd = jTextField50.getText();
        String ed = jTextField51.getText();
        String proc = "call
addIt('"+at+"', '"+fn+"', '"+ln+"', '"+sal+"', '"+bCode+"', '"+pass+"', '"+sd+"', '"+ed+"')";
        System.out.println(proc);
        try{
            Connection con = ConnectionToDb.connectMySqlDB();
            CallableStatement stmt = con.prepareCall(proc);
            ResultSet rs = stmt.executeQuery();

            DefaultTableModel itOfficerModel =
(DefaultTableModel)jTable15.getModel();
            String tbData[] = {at, fn, ln, sal, bCode, pass, sd, ed};
            itOfficerModel.addRow(tbData);

            con.close();
        }catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}

    }//GEN-LAST:event_jButton3ActionPerformed

    private void jButton44ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jButton44ActionPerformed
    // TODO add your handling code here:

    int selectedRow = jTable15.getSelectedRow();
    String it_AT = jTable15.getValueAt(selectedRow, 0).toString();

    String deleteCommand = "delete from itOfficer where it_AT= '" + it_AT
+ "';";

    try {
        Connection con = ConnectionToDb.connectMySqlDB();

        Statement stmt=con.createStatement();
        stmt.executeUpdate(deleteCommand);

        DefaultTableModel tbModel =
(DefaultTableModel)jTable15.getModel();

        tbModel.removeRow(selectedRow);
        con.close();
    }
    catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
    }//GEN-LAST:event_jButton44ActionPerformed

    private void jButton43ActionPerformed(java.awt.event.ActionEvent evt)

```



```

{ //GEN-FIRST:event_jButton43ActionPerformed
    // TODO add your handling code here:
    int selectedRow = jTable15.getSelectedRow();
    if(selectedRow != -1){

        String key = jTable15.getValueAt(selectedRow, 0).toString();

        try {

            Connection con = ConnectionToDb.connectMySQLDB();

            String[] data = new String[ 4 ];

            data[0] = jComboBox17.getSelectedItem().toString();
            data[1] = jTextField38.getText();
            data[2] = jTextField39.getText();
            data[3] = jTextField40.getText();

            String command0 = "UPDATE itOfficer set it_AT='" + data[0] +
            "' where it_AT= '" + key + "' ";
            String command1 = "UPDATE itOfficer set password= '" +
            data[1] + "' where it_AT= '" + key + "' ";
            String command2 = "UPDATE itOfficer set start_date = '" +
            data[2] + "' where it_AT= '" + key + "' ";
            String command3 = "UPDATE itOfficer set end_date = '" +
            data[3] + "' where it_AT= '" + key + "' ";

            if(!data[1].equals("")){
                CallableStatement stmt = con.prepareCall(command1);
                stmt.executeUpdate();
            }
            if(!data[2].equals("")){
                CallableStatement stmt = con.prepareCall(command2);
                stmt.executeUpdate();
            }
            if(!data[3].equals("")){
                CallableStatement stmt = con.prepareCall(command3);
                stmt.executeUpdate();
            }
            if(!data[0].equals("")){
                CallableStatement stmt = con.prepareCall(command0);
                stmt.executeUpdate();
            }

            //Update table
            CallableStatement stmt = con.prepareCall("select * from
itOfficer");

            ResultSet rs= stmt.executeQuery();
            DefaultTableModel tbModel =
(DefaultTableModel)jTable15.getModel();
            tbModel.setRowCount(0);
            while(rs.next()){

                String it_AT = rs.getString("it_AT") ;
                String password = rs.getString("password") ;
                String start_date = rs.getString("start_date");

```

```

        String end_date = rs.getString("end_date");

        String tbData[] = {it_AT, password, start_date , end_date
};

        tbModel.addRow(tbData);
    }

    con.close();
}
catch(Exception e){ JFrame f=new JFrame();
JOptionPane.showMessageDialog(f,e,"Alert",JOptionPane.WARNING_MESSAGE);}
}
} //GEN-LAST:event_jButton43ActionPerformed

private void jTextField50ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jTextField50ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jTextField50ActionPerformed

private void jComboBox5ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jComboBox5ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jComboBox5ActionPerformed

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting
code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {
        java.util.logging.Logger.getLogger(Menu.class.getName()).log(java.util.logging.
Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {
        java.util.logging.Logger.getLogger(Menu.class.getName()).log(java.util.logging.
Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

```

```

java.util.logging.Logger.getLogger(Menu.class.getName()).log(java.util.logging.
Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Menu.class.getName()).log(java.util.logging
Level.SEVERE, null, ex);
    }
    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Menu().setVisible(true);
        }
    });
}

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JPanel GuideMenuPanel;
private javax.swing.JPanel adminMenuPanel;
private javax.swing.JPanel adminsPanel;
private javax.swing.JPanel availableTripsPanel;
private javax.swing.JPanel branchInfoPanel;
private javax.swing.JPanel branchesMenuPanel;
private javax.swing.JPanel branchesPanel;
private javax.swing.JPanel dataMenuPanel;
private javax.swing.JPanel destinationsPanel;
private javax.swing.JPanel driverMenuPanel;
private javax.swing.JPanel driversPanel;
private javax.swing.JTextField endDateTextField1;
private javax.swing.JPanel eventsPanel;
private javax.swing.JPanel guidesPanel;
private javax.swing.JPanel itOfficersPanel;
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton10;
private javax.swing.JButton jButton11;
private javax.swing.JButton jButton12;
private javax.swing.JButton jButton13;
private javax.swing.JButton jButton14;
private javax.swing.JButton jButton15;
private javax.swing.JButton jButton16;
private javax.swing.JButton jButton17;
private javax.swing.JButton jButton18;
private javax.swing.JButton jButton19;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton20;
private javax.swing.JButton jButton21;
private javax.swing.JButton jButton22;
private javax.swing.JButton jButton23;
private javax.swing.JButton jButton24;
private javax.swing.JButton jButton25;
private javax.swing.JButton jButton26;
private javax.swing.JButton jButton27;
private javax.swing.JButton jButton28;
private javax.swing.JButton jButton29;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton30;

```

[illegible]


```
private javax.swing.JLabel jLabel88;
private javax.swing.JLabel jLabel9;
private javax.swing.JLabel jLabel90;
private javax.swing.JLabel jLabel91;
private javax.swing.JLabel jLabel92;
private javax.swing.JLabel jLabel93;
private javax.swing.JLabel jLabel94;
private javax.swing.JLabel jLabel95;
private javax.swing.JLabel jLabel96;
private javax.swing.JLabel jLabel97;
private javax.swing.JLabel jLabel98;
private javax.swing.JLabel jLabel99;
private javax.swing.JMenu jMenu1;
private javax.swing.JMenu jMenu2;
private javax.swing.JMenu jMenu3;
private javax.swing.JMenu jMenu4;
private javax.swing.JMenu jMenu5;
private javax.swing.JMenu jMenu6;
private javax.swing.JMenuBar jMenuBar1;
private javax.swing.JMenuItem jMenuItem1;
private javax.swing.JMenuItem jMenuItem10;
private javax.swing.JMenuItem jMenuItem11;
private javax.swing.JMenuItem jMenuItem12;
private javax.swing.JMenuItem jMenuItem13;
private javax.swing.JMenuItem jMenuItem14;
private javax.swing.JMenuItem jMenuItem15;
private javax.swing.JMenuItem jMenuItem16;
private javax.swing.JMenuItem jMenuItem17;
private javax.swing.JMenuItem jMenuItem18;
private javax.swing.JMenuItem jMenuItem19;
private javax.swing.JMenuItem jMenuItem2;
private javax.swing.JMenuItem jMenuItem20;
private javax.swing.JMenuItem jMenuItem21;
private javax.swing.JMenuItem jMenuItem22;
private javax.swing.JMenuItem jMenuItem23;
private javax.swing.JMenuItem jMenuItem3;
private javax.swing.JMenuItem jMenuItem4;
private javax.swing.JMenuItem jMenuItem5;
private javax.swing.JMenuItem jMenuItem6;
private javax.swing.JMenuItem jMenuItem7;
private javax.swing.JMenuItem jMenuItem8;
private javax.swing.JMenuItem jMenuItem9;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JScrollPane jScrollPane10;
private javax.swing.JScrollPane jScrollPane11;
private javax.swing.JScrollPane jScrollPane12;
private javax.swing.JScrollPane jScrollPane13;
private javax.swing.JScrollPane jScrollPane14;
private javax.swing.JScrollPane jScrollPane15;
private javax.swing.JScrollPane jScrollPane16;
private javax.swing.JScrollPane jScrollPane17;
private javax.swing.JScrollPane jScrollPane18;
private javax.swing.JScrollPane jScrollPane19;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JScrollPane jScrollPane20;
private javax.swing.JScrollPane jScrollPane21;
private javax.swing.JScrollPane jScrollPane22;
```

```
private javax.swing.JScrollPane jScrollPane23;
private javax.swing.JScrollPane jScrollPane24;
private javax.swing.JScrollPane jScrollPane25;
private javax.swing.JScrollPane jScrollPane26;
private javax.swing.JScrollPane jScrollPane27;
private javax.swing.JScrollPane jScrollPane28;
private javax.swing.JScrollPane jScrollPane29;
private javax.swing.JScrollPane jScrollPane3;
private javax.swing.JScrollPane jScrollPane4;
private javax.swing.JScrollPane jScrollPane5;
private javax.swing.JScrollPane jScrollPane6;
private javax.swing.JScrollPane jScrollPane7;
private javax.swing.JScrollPane jScrollPane8;
private javax.swing.JScrollPane jScrollPane9;
private javax.swing.JSeparator jseparator1;
private javax.swing.JSeparator jseparator2;
private javax.swing.JSeparator jseparator3;
private javax.swing.JSeparator jseparator4;
private javax.swing.JSeparator jseparator5;
private javax.swing.JSeparator jseparator6;
private javax.swing.JSeparator jseparator7;
private javax.swing.JSeparator jseparator8;
private javax.swing.JSeparator jseparator9;
private javax.swing.JTable jTable1;
private javax.swing.JTable jTable10;
private javax.swing.JTable jTable11;
private javax.swing.JTable jTable12;
private javax.swing.JTable jTable13;
private javax.swing.JTable jTable14;
private javax.swing.JTable jTable15;
private javax.swing.JTable jTable16;
private javax.swing.JTable jTable17;
private javax.swing.JTable jTable18;
private javax.swing.JTable jTable19;
private javax.swing.JTable jTable2;
private javax.swing.JTable jTable20;
private javax.swing.JTable jTable21;
private javax.swing.JTable jTable22;
private javax.swing.JTable jTable23;
private javax.swing.JTable jTable24;
private javax.swing.JTable jTable25;
private javax.swing.JTable jTable26;
private javax.swing.JTable jTable27;
private javax.swing.JTable jTable3;
private javax.swing.JTable jTable4;
private javax.swing.JTable jTable5;
private javax.swing.JTable jTable6;
private javax.swing.JTable jTable7;
private javax.swing.JTable jTable8;
private javax.swing.JTable jTable9;
private javax.swing.JTextArea jTextArea1;
private javax.swing.JTextArea jTextArea2;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField10;
private javax.swing.JTextField jTextField11;
private javax.swing.JTextField jTextField12;
private javax.swing.JTextField jTextField13;
```



```
private javax.swing.JTextField jTextField14;
private javax.swing.JTextField jTextField15;
private javax.swing.JTextField jTextField16;
private javax.swing.JTextField jTextField17;
private javax.swing.JTextField jTextField18;
private javax.swing.JTextField jTextField19;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField20;
private javax.swing.JTextField jTextField21;
private javax.swing.JTextField jTextField22;
private javax.swing.JTextField jTextField23;
private javax.swing.JTextField jTextField24;
private javax.swing.JTextField jTextField25;
private javax.swing.JTextField jTextField26;
private javax.swing.JTextField jTextField27;
private javax.swing.JTextField jTextField28;
private javax.swing.JTextField jTextField29;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField30;
private javax.swing.JTextField jTextField31;
private javax.swing.JTextField jTextField32;
private javax.swing.JTextField jTextField33;
private javax.swing.JTextField jTextField34;
private javax.swing.JTextField jTextField35;
private javax.swing.JTextField jTextField36;
private javax.swing.JTextField jTextField37;
private javax.swing.JTextField jTextField38;
private javax.swing.JTextField jTextField39;
private javax.swing.JTextField jTextField4;
private javax.swing.JTextField jTextField40;
private javax.swing.JTextField jTextField41;
private javax.swing.JTextField jTextField42;
private javax.swing.JTextField jTextField43;
private javax.swing.JTextField jTextField44;
private javax.swing.JTextField jTextField45;
private javax.swing.JTextField jTextField46;
private javax.swing.JTextField jTextField47;
private javax.swing.JTextField jTextField49;
private javax.swing.JTextField jTextField5;
private javax.swing.JTextField jTextField50;
private javax.swing.JTextField jTextField51;
private javax.swing.JTextField jTextField6;
private javax.swing.JTextField jTextField7;
private javax.swing.JTextField jTextField8;
private javax.swing.JTextField jTextField9;
private javax.swing.JPanel languagesPanel;
private javax.swing.JPanel logsMenuPanel;
private javax.swing.JPanel managersPanel;
private javax.swing.JPanel offersMenuPanel;
private javax.swing.JPanel offersPanel;
private javax.swing.JPanel phonesPanel;
private javax.swing.JPanel reservationOffersPanel;
private javax.swing.JPanel reservationsPanel;
private javax.swing.JTextField startDateTextField1;
private javax.swing.JPanel travelToPanel;
private javax.swing.JPanel tripsPanel;
private javax.swing.JPanel workersPanel;
```

```
// End of variables declaration//GEN-END:variables  
}
```