

Project Εργαστηρίου Βάσεων Δεδομένων

Περιεχόμενα

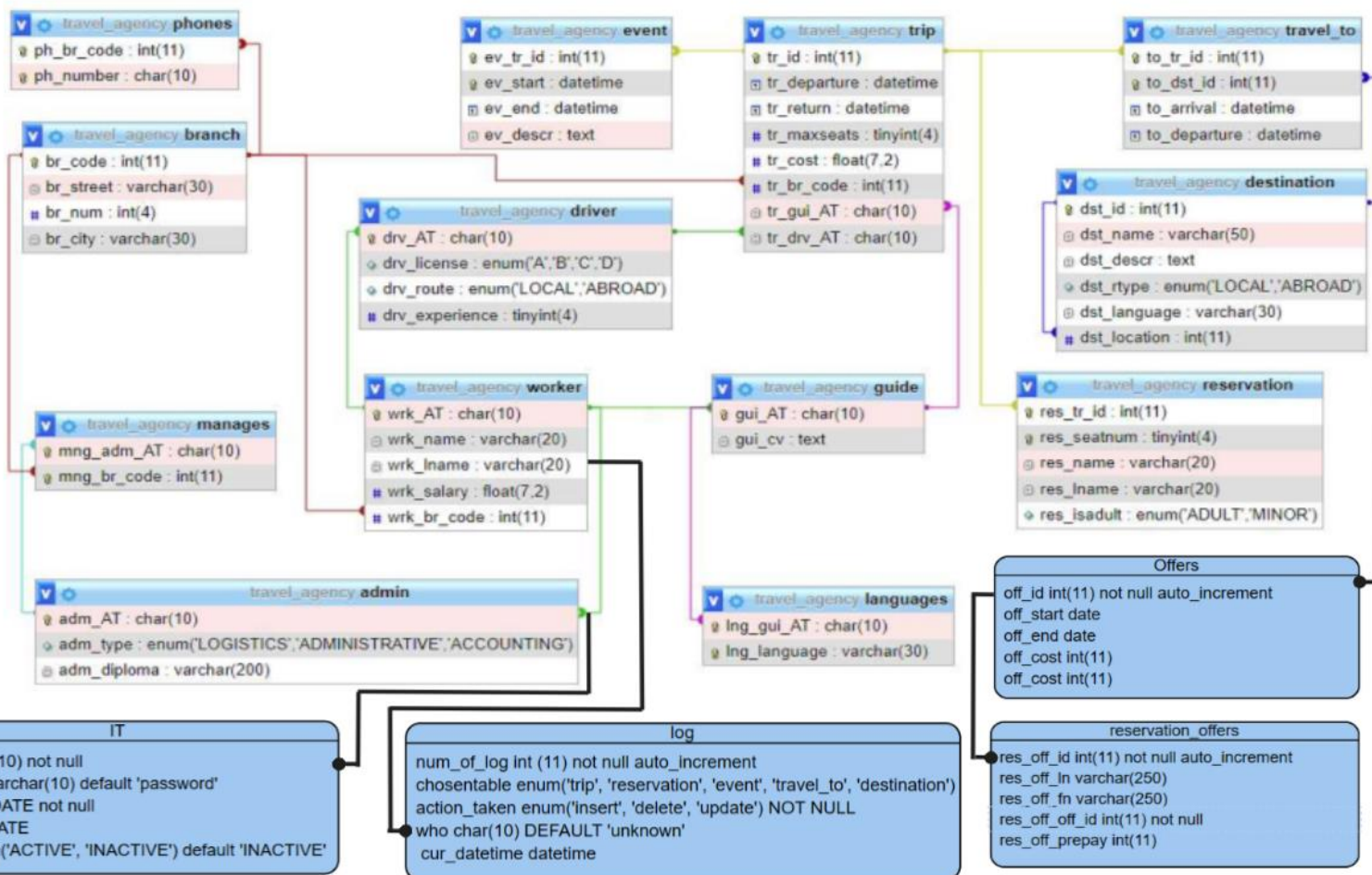
1.Παραδοχές.....	3
2.Περιγραφή της Βάσης Δεδομένων & Διάγραμμα Βάσης.....	4
3. Περιεχόμενο Εργασίας	6
3.1 Μέρος Α: Σχεδιασμός ΒΔ και SQL.....	6
3.1.2 Δημιουργία Stored Procedure.....	6
3.1.3 Δημιουργία Trigger.....	14
3.2 Μέρος Β: GUI.....	23
4. Σενάριο.....	27
5. Extra λειτουργία.....	35
6. Κώδικες.....	37

1. ΠΑΡΑΔΟΧΕΣ

1. trigger_reservation: δημιουργήσαμε επιπλέον trigger από αυτούς που ζητούνται ώστε όταν γίνεται κράτηση στον πίνακα reservation τότε το πεδίο seatnum να παίρνει τιμή η οποία κάθε φορά θα είναι η επόμενη θέση από την προηγούμενη κράτηση. Κατά αυτόν τον τρόπο οι θέσεις θα έχουν αύξουσα διαδοχική σειρά.
2. Πίνακας log: ορίσαμε επιπλέον πίνακα στη βάση για να καταγράφεται η δραστηριότητα των υπεύθυνων πληροφορικής της εταιρίας για τους πίνακες trip, reservation, event, travel_to, destination. Εξηγείται λεπτομερώς παρακάτω.
3. Οι εντολές insert για την δημιουργία της βάσης έχουν παραχθεί μέσω προγράμματος python. Το αρχείο βρίσκεται μέσα στο zip με την ονομασία database_final.py . Αν επιθυμείτε να το εκτελέσετε για να παράγετε τις insert εντολές θα πρέπει να αλλάξετε την τιμή της μεταβλητής path1. Θα πρέπει να επιλέξετε σε ποιο path θέλετε να παραχθεί το αρχείο και στο ίδιο path που θα επιλέξετε για να παραχθεί θα πρέπει να υπάρχει το αρχείο names.txt (αυτό είναι το txt που μας δίνετε το οποίο περιέχει ονόματα).
4. Παραλείψαμε τις εντολές insert στον πίνακα reservation_offers λόγω χώρου.

2.Περιγραφή της Βάσης Δεδομένων & Διάγραμμα Βάσης

Σχεσιακό διάγραμμα της συνολικής αναθεωρημένης ΒΔ



Προσθέσαμε 4 νέους πίνακες:

1. Offers: προσφορές ταξιδιών
2. Reservation_offers: οι κρατήσεις για τις προσφορές των ταξιδιών
3. IT: υπεύθυνος πληροφορικής του πρακτορείου, έχουμε προσθέσει πεδίο με ονομασία active που δηλώνει το status το υπεύθυνου πληροφορικής ώστε να γνωρίζουμε κάθε φορά ποιος IT κάνει τις αλλαγές στη βάση και να καταχωρούμε το όνομά του στον πίνακα log.
4. Log: πίνακας που αποθηκεύεται όλη η δραστηριότητα. Κάθε ενέργεια εισαγωγής, ενημέρωσης ή διαγραφής στους πίνακες trip, reservation, event, travel_to, destination, καθώς και το ποιος την εκτέλεσε.

Για τις ανάγκες του project επιλέξαμε να προσθέσουμε επιπλέον τον πίνακα log με τον οποίο αποθηκεύουμε τις μετατροπές που κάνουν οι υπεύθυνοι πληροφορικής του πρακτορείου. Ο πίνακας log αποτελείται από τα εξής πεδία:

- num_of_log int (11) not null auto_increment: αριθμός ο οποίος δίνεται σε κάθε ενέργεια και είναι μοναδικός, αυξάνεται αυτόματα και είναι το κλειδί του πίνακα.
- chosentable enum('trip', 'reservation', 'event', 'travel_to', 'destination'): το πεδίο δηλώνει τον πίνακα στον οποίο έκανε αλλαγή ο υπεύθυνος πληροφορικής(IT).
- action_taken enum('insert', 'delete', 'update') NOT NULL: δηλώνει το είδος της ενέργειας.
- who char(10) DEFAULT 'unknown': ποιος έκανε την ενέργεια, το επώνυμο του υπεύθυνου πληροφορικής που την έκανε.
- cur_datetime datetime: η τρέχουσα ώρα και ημέρα πραγματοποίηση της ενέργειας.

Τα πεδία των υπόλοιπων πινάκων έχουν σχεδιαστεί σύμφωνα με την εκφώνηση και δεν έχουμε κάνει καμία προσθήκη δική μας.

Οι εντολές create και insert για τη δημιουργία της βάσης βρίσκονται στα αρχεία του zip με ονομασία “1.create table.txt” και “3.automation.txt” αντίστοιχα και στο τέλος της αναφοράς τα παραθέτουμε όλα για οικονομία χώρου.

3. Περιεχόμενο Εργασίας

3.1.2 Δημιουργία Stored Procedure

- Ερώτημα 3.1.3.1

drop procedure if exists InsertDriver;

delimiter \$

```
create procedure InsertDriver(in AT char(10), in name varchar(20), in lastname varchar(20), in salary float(7,2),  
                             in license enum('A','B','C','D'), in route enum('LOCAL','ABROAD'), in experience tinyint(4))
```

BEGIN

```
declare branch_code int(11);
```

```
SELECT wrk_br_code into branch_code from worker inner join driver on wrk_AT = drv_AT GROUP BY  
wrk_br_code ORDER BY count(*) limit 1;
```

```
insert into worker values(AT, name, lastname, salary, branch_code);
```

```
insert into driver values(AT, license, route, experience);
```

END\$

DELIMITER ;

```
call InsertDriver('AM2754238','Giorgos','Samios', 5000, 'B', 'LOCAL', 35);
```

Στην παρακάτω εικόνα είναι ένα παράδειγμα για την κλήση της procedure InsertDriver:

Κάνουμε select στην ένωση των πινάκων driver και worker για να εμφανίσει πόσους οδηγούς έχει κάθε επιχείρηση και να βρούμε αυτή με τους λιγότερους. Έπειτα καλούμε την procedure που δημιουργήσαμε και ο οδηγός που έχουμε εισάγει βρίσκεται στην επιχείρηση με τους λιγότερους.

```
mysql> SELECT wrk_br_code, count(*) from worker inner join driver on wrk_AT = drv_AT GROUP BY wrk_br_code;
+-----+-----+
| wrk_br_code | count(*) |
+-----+-----+
|          6 |         2 |
|          8 |         2 |
|          7 |         2 |
|          3 |         2 |
|          9 |         2 |
|          5 |         2 |
|          4 |         2 |
|          2 |         2 |
|         10 |         1 |
+-----+-----+
9 rows in set (0.00 sec)

mysql> call InsertDriver('AM2754237','Giorgos','Samios', 5000, 'B', 'LOCAL', 35);
Query OK, 1 row affected (0.01 sec)

mysql> SELECT wrk_br_code, count(*) from worker inner join driver on wrk_AT = drv_AT GROUP BY wrk_br_code;
+-----+-----+
| wrk_br_code | count(*) |
+-----+-----+
|          6 |         2 |
|          8 |         2 |
|          7 |         2 |
|          3 |         2 |
|          9 |         2 |
|          5 |         2 |
|          4 |         2 |
|         10 |         2 |
|          2 |         2 |
+-----+-----+
9 rows in set (0.00 sec)
```

- Ερώτημα 3.1.3.2

drop procedure if exists SearchTrip;

delimiter \$

create procedure SearchTrip(in id int, in startdate datetime, in enddate datetime)

BEGIN

```
CREATE TEMPORARY TABLE temp1 (tr_id1 INT,tr_cost INT,tr_maxseats INT,  
    wrk_name varchar(20), wrk_lname varchar(20), tr_departure DATETIME,  
    tr_return DATETIME);
```

```
CREATE TEMPORARY TABLE temp2 (tr_id2 INT,numofreservations INT,availableseats INT);
```

```
insert into temp1 select tr_id as trip1, tr_cost,tr_maxseats,wrk_name,wrk_lname,  
tr_departure, tr_return
```

```
from branch inner join trip on tr_br_code=br_code
```

```
    inner join driver on drv_AT=tr_drv_AT
```

```
    inner join guide on tr_gui_AT=gui_AT
```

```
    inner join worker on wrk_AT=gui_AT or wrk_AT=drv_AT
```

```
where br_code=id and startdate>=tr_departure and startdate<tr_return
```

```
and enddate>tr_departure and enddate<=tr_return;
```

```
insert into temp2 select tr_id as trip2, count(*) as numofreservations , tr_maxseats-  
count(*) as availableseats
```

```
from branch
```

```
    inner join trip on br_code=tr_br_code
```

```
    inner join reservation on tr_id=res_tr_id
```

```
where br_code=id
```

```
group by res_tr_id;
```

```
select * from temp1 inner join temp2 on tr_id1=tr_id2;
```

```
drop table temp1;
```



```
drop table temp2;
```

```
END$
```

```
DELIMITER ;
```

```
call SearchTrip(4, '2022-11-13 09:00:00', '2022-12-16 12:00:00');
```

Δίνουμε ως όρισμα τον αριθμό του υποκαταστήματος και 2 ημερομηνίες και επιστρέφονται πληροφορίες για το ταξίδι:

Κωδικός ταξιδιού, κόστος, μέγιστες θέσεις, όνομα ξεναγού και οδηγού, επώνυμο ξεναγού και οδηγού, ημερομηνία αναχώρησης και επιστροφής, κωδικός ταξιδιού ξανά, αριθμός κρατήσεων για κάθε ταξίδι και διαθέσιμες θέσεις.

(τα δύο τελευταία στοιχεία αναφέρονται δύο φορές διότι για τις ανάγκες του gui έπρεπε να τα συμπτύξουμε όλα σε ένα πίνακα)

```
call SearchTrip(4, '2022-11-13 09:00:00', '2022-12-16 12:00:00');
```

tr_id1	tr_cost	tr_maxseats	wrk_name	wrk_lname	tr_departure	tr_return	tr_id2	numofreservations	availableseats
4	900	33	Aaren	White	2022-11-13 09:00:00	2022-12-16 12:00:00	4	10	23
4	900	33	Lachish	Davis	2022-11-13 09:00:00	2022-12-16 12:00:00	4	10	23
14	900	27	Aaren	White	2022-11-13 09:00:00	2022-12-16 12:00:00	14	10	17
14	900	27	Lachish	Davis	2022-11-13 09:00:00	2022-12-16 12:00:00	14	10	17

```
4 rows in set (0.00 sec)
```

```
Query OK, 0 rows affected (0.02 sec)
```

- Ερώτημα 3.1.3.3

```
drop procedure if exists deleteAdmin;

delimiter $

create procedure deleteAdmin(in name varchar(20), in lname varchar(20))

BEGIN

    DECLARE type enum('LOGISTICS','ADMINISTRATIVE','ACCOUNTING');

    select adm_type into type
    from worker
    inner join admin on wrk_AT=adm_AT
    where wrk_name=name and wrk_lname=lname;

    if (type = 'ADMINISTRATIVE') then

        SIGNAL SQLSTATE VALUE '45000'

        SET MESSAGE_TEXT = 'CAN NOT DELETE ADMIN.';

    end if;

delete from worker where wrk_name=name and wrk_lname=lname;

END$

DELIMITER ;

call deleteAdmin( 'Nadiya', 'Thompson');
```

Δίνουμε ως όρισμα το όνομα και το επώνυμο ενός υπαλλήλου ο οποίος είναι διευθυντής και δεν τον διαγράφει ενώ όταν δώσουμε το όνομα και το επώνυμο ενός υπαλλήλου που είναι διοικητικός τότε τον διαγράφει.

```
mysql> call deleteAdmin( 'Nadiya', 'Thompson');
ERROR 1644 (45000): CAN NOT DELETE ADMIN.
mysql> |
```

```
mysql> call deleteAdmin( 'Creigh', 'Moore');  
Query OK, 1 row affected (0.01 sec)
```

- Ερώτημα 3.1.3.4.α

drop procedure if exists customerPrepay;

delimiter \$

create procedure customerPrepay (in minimumvalue int, in maximumvalue int)

BEGIN

select res_off_ln, res_off_fn from reservation_offers

where res_off_prepay BETWEEN minimumvalue and maximumvalue;

END\$

DELIMITER ;

call customerPrepay(50,200);

Δημιουργία INDEXING για το procedure customerPrepay:

CREATE INDEX Prepay USING HASH ON reservation_offers (res_off_prepay);

DROP INDEX Prepay ON reservation_offers;

Αποτέλεσμα χωρίς index call customerPrepay(50,53);

Russell	Alisa
Wong	Arielle
Bates	Allyson
Willis	Rylan
Lloyd	Alessandra
Sandoval	Allie
Abbott	Moshe
Sharp	Humberto
Brock	Ashleigh

1600 rows in set (0.03 sec)

Query OK, 0 rows affected (0.97 sec)

Αποτέλεσμα με index call customerPrepay(50,53);

Shaw	Sullivan
Cooley	Lucia
Dorsey	Belinda
Mathis	Carter
Olsen	Kelsey
Terrell	Sophie
Lloyd	Alessandra
Sharp	Humberto

1600 rows in set (0.01 sec)

Query OK, 0 rows affected (0.97 sec)

- Ερώτημα 3.1.3.4.β

drop procedure if exists customerReservation;

delimiter \$

create procedure customerReservation (in lastname varchar(250))

BEGIN

select res_off_ln, res_off_fn, res_off_off_id, count(*) from reservation_offers

where res_off_ln=lastname group by res_off_off_id;

END\$

DELIMITER ;

call customerReservation('Ruiz');

Δημιουργία INDEXING για το procedure customerReservation:

CREATE INDEX Reservations USING HASH ON reservation_offers (res_off_ln);

DROP INDEX Reservations ON reservation_offers;

```
mysql> call customerReservation('Higgins');
+-----+-----+-----+-----+
| res_off_ln | res_off_fn | res_off_off_id | count(*) |
+-----+-----+-----+-----+
| Higgins   | Dominik   | 1              | 25       |
| Higgins   | Jayson    | 2              | 34       |
| Higgins   | Armani    | 3              | 31       |
+-----+-----+-----+-----+
3 rows in set (0.12 sec)

Query OK, 0 rows affected (0.12 sec)

mysql> CREATE INDEX Reservations USING HASH ON reservation_offers (res_off_ln);
Query OK, 0 rows affected, 1 warning (0.32 sec)
Records: 0 Duplicates: 0 Warnings: 1

mysql> call customerReservation('Higgins');
+-----+-----+-----+-----+
| res_off_ln | res_off_fn | res_off_off_id | count(*) |
+-----+-----+-----+-----+
| Higgins   | Dominik   | 1              | 25       |
| Higgins   | Jayson    | 2              | 34       |
| Higgins   | Armani    | 3              | 31       |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)
```

3.1.3 Δημιουργία Trigger

- **Ερώτημα 3.1.4.1**

```
DROP TRIGGER IF EXISTS triggerTripInsert;
```

```
DELIMITER $
```

```
CREATE TRIGGER triggerTripInsert AFTER INSERT ON trip
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare lastname varchar(20);
```

```
    select wrk_lname into lastname from worker inner join IT on wrk_AT = IT_AT
```

```
    where active='ACTIVE';
```

```
    insert into log values(NULL,'trip', 'insert', lastname, CURRENT_TIMESTAMP());
```

```
END$
```

```
DELIMITER ;
```

```
DROP TRIGGER IF EXISTS triggerTripDelete;
```

```
DELIMITER $
```

```
CREATE TRIGGER triggerTripDelete AFTER DELETE ON trip
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare lastname varchar(20);
```

```
    select wrk_lname into lastname from worker inner join IT on wrk_AT = IT_AT
```

```
    where active='ACTIVE';
```

```
    insert into log values(NULL,'trip', 'delete', lastname, CURRENT_TIMESTAMP());
```

```
END$
```

```
DELIMITER ;
```

```
DROP TRIGGER IF EXISTS triggerTripUpdate;
```

```
DELIMITER $
```

```
CREATE TRIGGER triggerTripUpdate AFTER UPDATE ON trip
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare lastname varchar(20);
```

```
    select wrk_lname into lastname from worker inner join IT on wrk_AT = IT_AT
```

```
    where active='ACTIVE';
```

```
    insert into log values(NULL,'trip', 'update', lastname, CURRENT_TIMESTAMP());
```

```
END$
```

```
DELIMITER ;
```

```
DROP TRIGGER IF EXISTS triggerReservationInsert;
```

```
DELIMITER $
```

```
CREATE TRIGGER triggerReservationInsert AFTER INSERT ON reservation
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare lastname varchar(20);
```

```
    select wrk_lname into lastname from worker inner join IT on wrk_AT = IT_AT
```

```
    where active='ACTIVE';
```

```
    insert into log values(NULL,'reservation', 'insert', lastname, CURRENT_TIMESTAMP());
```

```
END$
```

```
DELIMITER ;
```

```
DROP TRIGGER IF EXISTS triggerReservationDelete;
```

```
DELIMITER $
```

```
CREATE TRIGGER triggerReservationDelete AFTER DELETE ON reservation
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare lastname varchar(20);
```

```
    select wrk_lname into lastname from worker inner join IT on wrk_AT = IT_AT
```

```
    where active='ACTIVE';
```

```
    insert into log values(NULL,'reservation', 'delete', lastname, CURRENT_TIMESTAMP());
```

```
END$
```

```
DELIMITER ;
```

```
DROP TRIGGER IF EXISTS triggerReservationUpdate;
```

```
DELIMITER $
```

```
CREATE TRIGGER triggerReservationUpdate AFTER UPDATE ON reservation
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare lastname varchar(20);
```

```
    select wrk_lname into lastname from worker inner join IT on wrk_AT = IT_AT
```

```
    where active='ACTIVE';
```

```
    insert into log values(NULL,'reservation', 'update', lastname, CURRENT_TIMESTAMP());
```

```
END$
```

```
DELIMITER ;
```



```
DROP TRIGGER IF EXISTS triggerTravel_toInsert;
```

```
DELIMITER $
```

```
CREATE TRIGGER triggerTravel_toInsert AFTER INSERT ON travel_to
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare lastname varchar(20);
```

```
    select wrk_lname into lastname from worker inner join IT on wrk_AT = IT_AT
```

```
    where active='ACTIVE';
```

```
    insert into log values(NULL,'travel_to', 'insert', lastname, CURRENT_TIMESTAMP());
```

```
END$
```

```
DELIMITER ;
```

```
DROP TRIGGER IF EXISTS triggerTravel_toDelete;
```

```
DELIMITER $
```

```
CREATE TRIGGER triggerTravel_toDelete AFTER DELETE ON travel_to
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare lastname varchar(20);
```

```
    select wrk_lname into lastname from worker inner join IT on wrk_AT = IT_AT
```

```
    where active='ACTIVE';
```

```
    insert into log values(NULL,'travel_to', 'delete', lastname, CURRENT_TIMESTAMP());
```

```
END$
```

```
DELIMITER ;
```

```
DROP TRIGGER IF EXISTS triggerTravel_toUpdate;
```

```
DELIMITER $
```

```
CREATE TRIGGER triggerTravel_toUpdate AFTER UPDATE ON travel_to
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare lastname varchar(20);
```

```
    select wrk_lname into lastname from worker inner join IT on wrk_AT = IT_AT
```

```
    where active='ACTIVE';
```

```
    insert into log values(NULL,'travel_to', 'update', lastname, CURRENT_TIMESTAMP());
```

```
END$
```

```
DELIMITER ;
```

```
DROP TRIGGER IF EXISTS triggerDestinationInsert;
```

```
DELIMITER $
```

```
CREATE TRIGGER triggerDestinationInsert AFTER INSERT ON destination
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare lastname varchar(20);
```

```
    select wrk_lname into lastname from worker inner join IT on wrk_AT = IT_AT
```

```
    where active='ACTIVE';
```

```
    insert into log values(NULL,'destination', 'insert', lastname, CURRENT_TIMESTAMP());
```

```
END$
```

```
DELIMITER ;
```

```
DROP TRIGGER IF EXISTS triggerDestinationDelete;
```

```
DELIMITER $
```

```
CREATE TRIGGER triggerDestinationDelete AFTER DELETE ON destination
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare lastname varchar(20);
```

```
    select wrk_lname into lastname from worker inner join IT on wrk_AT = IT_AT
```

```
    where active='ACTIVE';
```

```
    insert into log values(NULL,'destination', 'delete', lastname, CURRENT_TIMESTAMP());
```

```
END$
```

```
DELIMITER ;
```

```
DROP TRIGGER IF EXISTS triggerDestinationUpdate;
```

```
DELIMITER $
```

```
CREATE TRIGGER triggerDestinationUpdate AFTER UPDATE ON destination
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare lastname varchar(20);
```

```
    select wrk_lname into lastname from worker inner join IT on wrk_AT = IT_AT
```

```
    where active='ACTIVE';
```

```
    insert into log values(NULL,'destination', 'update', lastname, CURRENT_TIMESTAMP());
```

```
END$
```

```
DELIMITER ;
```

Πραγματοποιούμε select για τον πίνακα log ώστε να δούμε ότι έχουν καταγραφεί οι ενέργειες που έχουμε κάνει:

```
mysql> select * from log;
```

num_of_log	chosentable	action_taken	who	cur_datetime
1	trip	insert	Smith	2023-01-20 17:56:23
297	trip	update	Brown	2023-01-27 19:04:46
298	trip	delete	Brown	2023-01-27 19:04:50
299	trip	insert	Brown	2023-01-27 19:04:52
300	travel_to	update	Brown	2023-01-27 19:05:16
301	travel_to	delete	Brown	2023-01-27 19:05:19
302	travel_to	insert	Brown	2023-01-27 19:05:22
303	reservation	update	Brown	2023-01-27 19:05:49
304	reservation	delete	Brown	2023-01-27 19:05:55
305	reservation	insert	Brown	2023-01-27 19:05:58
306	destination	update	Brown	2023-01-27 19:06:15
307	destination	delete	Brown	2023-01-27 19:06:18
308	destination	insert	Brown	2023-01-27 19:06:20

- Ερώτημα 3.1.4.2

```
DROP TRIGGER IF EXISTS trigger2;
```

```
DELIMITER $
```

```
CREATE TRIGGER trigger2 BEFORE UPDATE ON trip
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    declare counter1 int;
```

```
    select count(*) into counter1 from reservation inner join trip on tr_id=res_tr_id
```

```
    where tr_id=new.tr_id group by tr_id;
```

```
    if (counter1>0 and (old.tr_departure<>new.tr_departure or old.tr_return<>new.tr_return  
    or old.tr_cost<>new.tr_cost)) then
```

```
        SIGNAL SQLSTATE VALUE '45000'
```

```
        SET MESSAGE_TEXT = 'CAN NOT UPDATE';
```

```
    end if;
```

```
END$
```

```
DELIMITER ;
```

Προσπαθούμε να αλλάξουμε το κόστος του ταξιδιού για το ταξίδι με κωδικό 4 και επειδή όπως βλέπουμε παρακάτω υπάρχουν κρατήσεις δεν μας επιτρέπει.

```
mysql> UPDATE trip
-> SET tr_cost=250
-> WHERE tr_id=4;
ERROR 1644 (45000): CAN NOT UPDATE
mysql> select * from reservation where res_tr_id=4;
```

res_tr_id	res_seatnum	res_name	res_lname	res_isadult
4	1	Lemcke	Mcdonald	ADULT
4	2	Zamir	Smith	ADULT
4	5	Fillander	Butler	MINOR
4	6	Henden	Olson	ADULT
4	7	Leander	Richardson	ADULT
4	8	Hyoza	Bailey	ADULT
4	9	Foskett	Bennett	MINOR
4	10	Shanleigh	Robertson	MINOR
4	11	Eisenstark	Lee	ADULT
4	12	Ulda	Chavez	ADULT

```
10 rows in set (0.00 sec)
```

- Ερώτημα 3.1.4.3

```
DROP TRIGGER IF EXISTS trigger3;
```

```
DELIMITER $
```

```
CREATE TRIGGER trigger3 BEFORE UPDATE ON worker
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    If (new.wrk_salary < old.wrk_salary) then
```

```
        SIGNAL SQLSTATE VALUE '45000'
```

```
        SET MESSAGE_TEXT = 'CAN NOT REDUCE WORKER SALARY!';
```

```
    end if;
```

```
END$
```

```
DELIMITER ;
```

Εμφανίζουμε αρχικά τον μισθό του υπαλλήλου με το παρακάτω AT και έπειτα κάνουμε ενημέρωση για τον συγκεκριμένο υπάλληλο με μισθό μικρότερο. Ο trigger δεν μας το επιτρέπει.

```
mysql> select * from worker where wrk_AT = 'AM2571561';
+-----+-----+-----+-----+-----+
| wrk_AT | wrk_name | wrk_lname | wrk_salary | wrk_br_code |
+-----+-----+-----+-----+-----+
| AM2571561 | Aarika | Thompson | 3000.00 | 2 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> update worker set wrk_salary = 2000.00 where wrk_AT = 'AM2571561';
ERROR 1644 (45000): CAN NOT REDUCE WORKER SALARY!
mysql> |
```

3.2 Μέρος B: GUI

- Ερώτημα 3.2.1 και 3.2.2

Περιγραφή gui:

public GetConnection(){...}

Η κλάση αυτή χρησιμοποιείται για να συνδεθούμε με τη βάση μας. Χρησιμοποιείται σε όλες τις κλάσεις που δημιουργούμε για να εκτελούμε τα queries. Θα πρέπει να έχουν συμπληρωθεί τα πεδία DB_URL, USERNAME, PASSWORD.

Κλάση LoginForm:

Ο χρήστης (IT) για να συνδεθεί στη βάση χρειάζεται να δώσει το όνομα και τον κωδικό του. Η λειτουργία αυτή επιτελείται στην κλάση LoginForm και αν τα στοιχεία που συμπλήρωσε ο χρήστης επαληθεύονται, τότε ο (IT) γίνεται ενεργός δηλαδή αποκτά το πεδίο status την τιμή active και καλείται η επόμενη κλάση Menu. Αν τα στοιχεία δεν επαληθεύονται εμφανίζεται κατάλληλο μήνυμα.

Κλάση Menu:

Η κλάση αυτή μας δίνει όλες τις επιλογές που έχει ο υπεύθυνος πληροφορικής. Επιλέγοντας μία από αυτές και το κουμπί select οδηγούμαστε στην κλάση που έχουμε επιλέξει. Η Menu δίνει τις ακόλουθες επιλογές:

1. **INSERT/UPDATE/DELETE:** Ο IT με αυτή την επιλογή μπορεί να κάνει αλλαγή σε όποιον πίνακα επιλέξει στο επόμενο βήμα. Επιλέγοντας INSERT/UPDATE/DELETE εμφανίζεται το frame της κλάσης ChooseTable.
2. **SearchTrip:** (3.1.3.2) Εμφανίζεται το frame της κλάσης SearchTrip. Η κλάση αυτή υλοποιεί σε γραφικό περιβάλλον την procedure SearchTrip.
3. **Customer's Reservation:** (3.1.3.4) Εμφανίζεται το frame της κλάσης CustomerReservation. Η κλάση αυτή υλοποιεί σε γραφικό περιβάλλον την procedure customerReservation.
4. **Branch's Infos:** Εμφανίζεται το frame της κλάσης BranchInfo και βλέπει τα στοιχεία κάθε υποκαταστήματος, το ονοματεπώνυμο του διευθυντή του, το σύνολο κρατήσεων και το σύνολο εσόδων.
5. **Branch's Workers:** Εμφανίζεται το frame της κλάσης BranchWorker και βλέπει για κάθε υποκατάστημα το όνομα, επώνυμο και μισθό όλων των υπάλληλων και το συνολικό ποσό μισθών που πληρώνει.
6. **Show Log:** Εμφανίζεται το frame της κλάσης ShowLog και βλέπει όλες τις ενέργειες που έχουν καταγραφεί στον πίνακα log.
7. **Revenue and Cost:** (BONUS) Εμφανίζει το frame της κλάσης RevenueandCost και βλέπει σύνολο εσόδων και εξόδων ανά υποκατάστημα.

Κλάση ChooseTable:

Ο IT θα πρέπει να επιλέξει σε ποιόν από τους 16 πίνακες της βάσης θέλει να κάνει αλλαγές (insert/update/delete). Δεν του δίνεται η δυνατότητα να κάνει αλλαγές μόνο στον πίνακα log.

Κλάση Branch:

Σε αυτή την κλάση ο IT μπορεί να εισάγει/ενημερώσει/διαγράψει. Στο frame που εμφανίζεται ο χρήστης βλέπει ένα πίνακα με όλες τις εγγραφές που περιέχει ο πίνακας branch της βάσης. Μπορεί με τον cursor να επιλέξει μια από αυτές τις εγγραφές που υπάρχουν, τότε οι τιμές αυτής της εγγραφής θα συμπληρώσουν τα πεδία που υπάρχουν δίπλα. Μπορεί ο IT να αλλάξει τις τιμές των πεδίων ώστε να ενημερώσει την εγγραφή ή να μην τα αλλάξει και να επιλέξει διαγραφή. Τέλος του δίνεται η δυνατότητα να συμπληρώσει τα πεδία ώστε να εισάγει μια εγγραφή. Για κάθε ενέργεια εμφανίζει κατάλληλο μήνυμα ώστε να γνωρίζουμε αν η ενέργεια ολοκληρώθηκε. Αν δεν επιθυμεί να κάνει κάποια άλλη ενέργεια μπορεί να επιστρέψει στο Menu.

Στον κώδικα της κλάσης δημιουργούμε λίστα στην οποία αποθηκεύονται τα αποτελέσματα του query που κάνει select όλα τα πεδία του πίνακα.

Τα κουμπιά INSERT, DELETE, UPDATE εκτελούν το query που τους έχουμε καθορίσει (είναι αντίστοιχο της λειτουργίας που δηλώνει η ονομασία) και ενημερώνουμε τη λίστα με τα νέα δεδομένα.

Κλάσεις Worker,Admin,Manages,Phones,Driver,Guide,Language,Trip,Event,Destination,Travel_to,Reservation,IT,Offers,Reservation_offers:

Για κάθε πίνακα της βάσης εκτός του log έχουμε δημιουργήσει αντίστοιχη κλάση με όνομα αυτό του πίνακα της βάσης. Σε κάθε μία από αυτές τις κλάσεις εμφανίζεται frame που να υπακούει στα πεδία του πίνακα που αντιστοιχεί και δίνεται η δυνατότητα στον IT να εισάγει, ενημερώσει, διαγράψει. Ο κώδικας με τον οποίο υλοποιούνται αυτές οι κλάσεις είναι ίδιος με της κλάσης Branch, διαφοροποιείται μόνο στον αριθμό των πεδίων αφού κάθε πίνακας της βάσης έχει διαφορετικό πλήθος πεδίων.

Επίσης κάθε μία από αυτές τις κλάσεις έχει δική της κλάση SetterClassname (Classname το όνομα της κλάσης στην οποία αναφέρεται ο setter) με την οποία δημιουργούμε setter για να αποθηκεύουμε τις τιμές.

Κλάση SearchTrip:

Στο frame που εμφανίζεται πρέπει ο IT να εισάγει στα πεδία τον κωδικό της εταιρίας, ημερομηνία αναχώρησης και επιστροφής. Για αυτά τα στοιχεία που δίνει στους πίνακες δίπλα εμφανίζει τις ακόλουθες πληροφορίες:

- Κωδικός ταξιδιού που ανήκει σε αυτό το υποκατάστημα
- Κόστος ταξιδιού
- Μέγιστος αριθμός θέσεων
- Όνομα οδηγού ή ξεναγού
- Επώνυμο οδηγού ή ξεναγού
- Ημερομηνία αναχώρησης
- Ημερομηνία επιστροφής
- Αριθμό κρατήσεων
- Διαθέσιμες θέσεις

Για να εκτελέσουμε αυτή τη λειτουργία καλούμε την procedure SearchTrip (3.1.3.2) και τα αποτελέσματα τα αποθηκεύουμε σε δύο λίστες. Αν στις ημερομηνίες που δόθηκαν δεν υπάρχει ταξίδι από το υποκατάστημα που επιλέχθηκε εμφανίζεται μήνυμα και μας ενημερώνει.

Κλάση CustomerReservation:

Σε αυτή την κλάση καλούμε την procedure customerReservation (3.1.3.4) και με βάση το επώνυμο που δόθηκε στο πεδίο του frame εμφανίζονται στον πίνακα τα ονόματα και επώνυμα των πελατών με το επώνυμο αυτό και την προσφορά ταξιδιού στην οποία έχει γίνει εγγραφή. Πατώντας το select ο χρήστης εκτελείται το query για την procedure με το όρισμα που δόθηκε, τα αποτελέσματα αποθηκεύονται σε λίστα και τα εμφανίζει ο πίνακας του frame.

Κλάση BranchInfo:

Εκτελούμε κατάλληλο query και τα αποτελέσματα εμφανίζονται στον πίνακα του frame. Στον πίνακα βλέπουμε για κάθε υποκατάστημα τα στοιχεία του, το ονοματεπώνυμο του διευθυντή του, το σύνολο κρατήσεων και το σύνολο εσόδων.

Query που εκτελούμε:

Μέσα σε ένα εσωτερικό select δημιουργούμε προσωρινό πίνακα ο οποίος μετά από join με τους πίνακες branch, trip, reservation έχει δεδομένα για κάθε υποκατάστημα: τον κωδικό του κάθε ταξιδιού, το κόστος του κάθε ταξιδιού και το σύνολο κρατήσεων (κάνουμε group by tr_id). Για το τελικό αποτέλεσμα κάνουμε join μεταξύ του προσωρινού πίνακα που δημιουργήσαμε και των πινάκων manages για να πάρουμε τον διευθυντή του κάθε υποκαταστήματος και του worker για να πάρουμε το ονοματεπώνυμο του διευθυντή. Κάνοντας στο τέλος group by br_code παίρνουμε το πλήθος των κρατήσεων για κάθε υποκατάστημα και το κέρδος.

Κλάση BranchWorker:

Εκτελούμε κατάλληλα queries και τα αποτελέσματα εμφανίζονται στους πίνακες του frame. Στους πίνακες βλέπουμε για κάθε υποκατάστημα το όνομα, επώνυμο και μισθό όλων των υπάλληλων του και το συνολικό ποσό μισθών που πληρώνει.

Queries που εκτελούμε:

Στο πρώτο query που εκτελούμε επιλέγουμε με join από τους πίνακες branch και worker το κωδικό της εταιρίας και τα στοιχεία των υπαλλήλων της ονοματεπώνυμο και μισθό.

Στο δεύτερο query εμφανίζεται το συνολικό ποσό μισθών κάθε υποκαταστήματος αφού επιλέγουμε group by br_code.

Κλάση ShowLog:

Εκτελούμε κατάλληλο query και τα αποτελέσματα εμφανίζονται στον πίνακα του frame. Στον πίνακα βλέπουμε όλα τα περιεχόμενα του πίνακα της βάσης log.

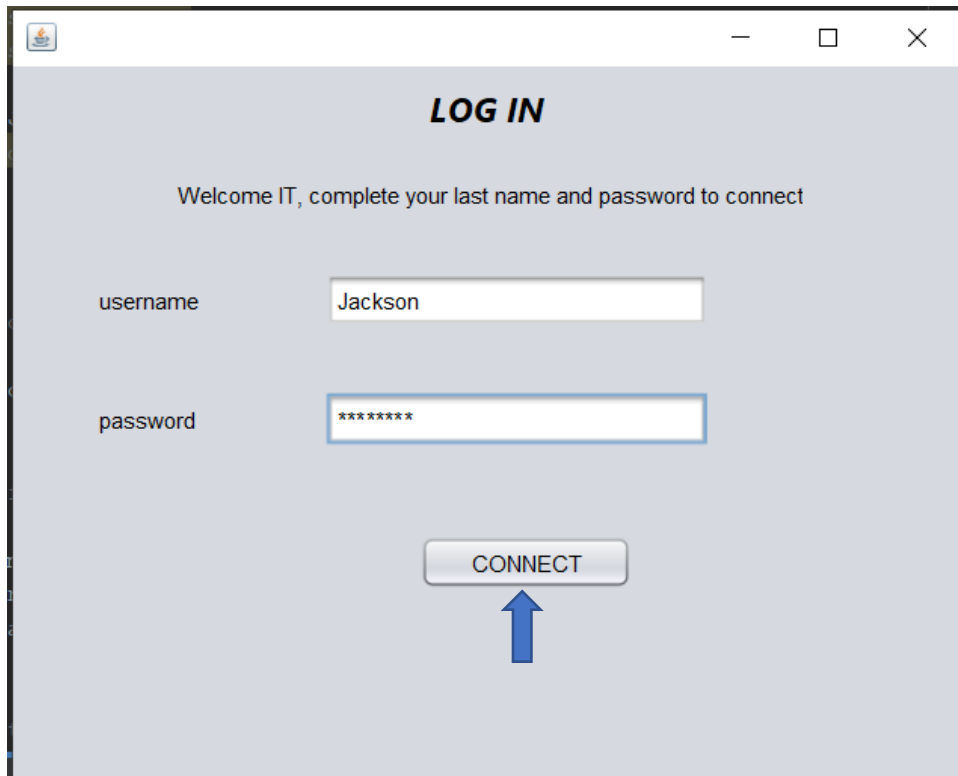
Query που εκτελούμε:

Κάνουμε select όλες τις στήλες του πίνακα log.

4. Σενάριο

Σενάριο: Θέλουμε να προσθέσουμε έναν υπάλληλο στο υποκατάστημα 2.

Κάνουμε log in με έναν IT.



The screenshot shows a web browser window with a login form. The window title is "LOG IN". The form has a heading "Welcome IT, complete your last name and password to connect". Below this, there are two input fields: "username" with the value "Jackson" and "password" with the value "*****". A "CONNECT" button is located below the password field, with a blue arrow pointing up to it.

Επιλέγουμε “Branch’s Workers” και πατάμε “SELECT”.

MENU

Choose an option to continue

☐ INSERT/UPDATE/DELETE

☐ Search Trip

☐ Customer's Reservation

☐ Branch's Infos

☒ Branch's Workers

☐ Show Log

☐ Revenue and Cost

SELECT

LOG OUT

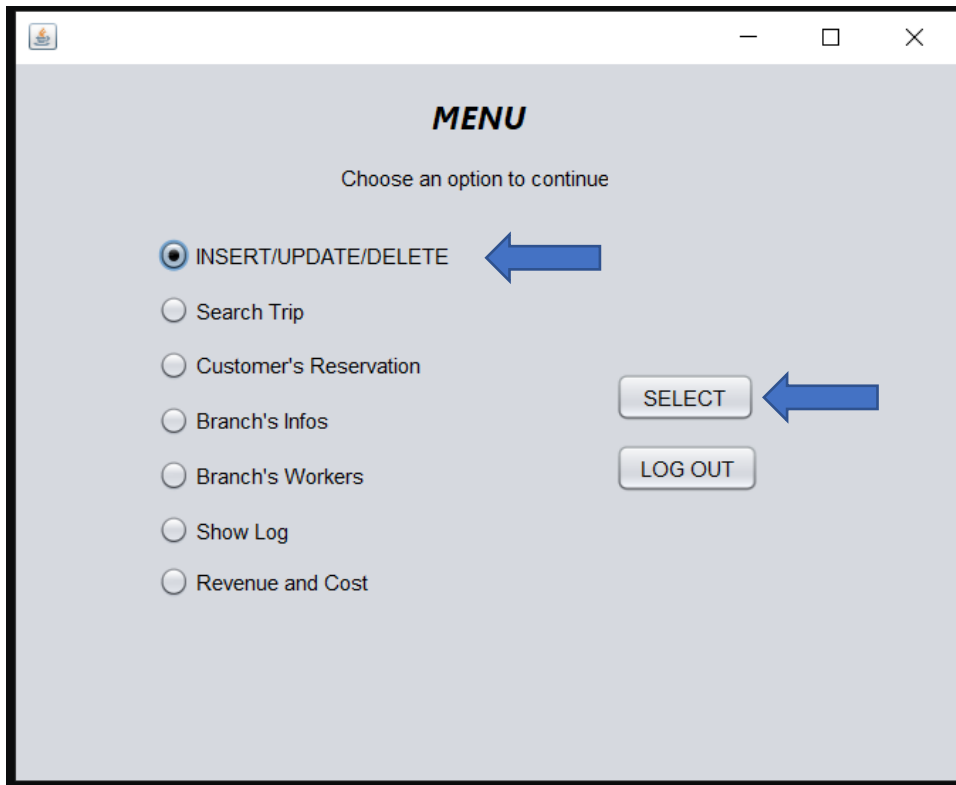
Βλέπουμε ότι το σύνολο των μισθών του υποκαταστήματος 2 είναι 14.000 και πατάμε “MENU” ώστε να γυρίσουμε στο μενού.

Menu

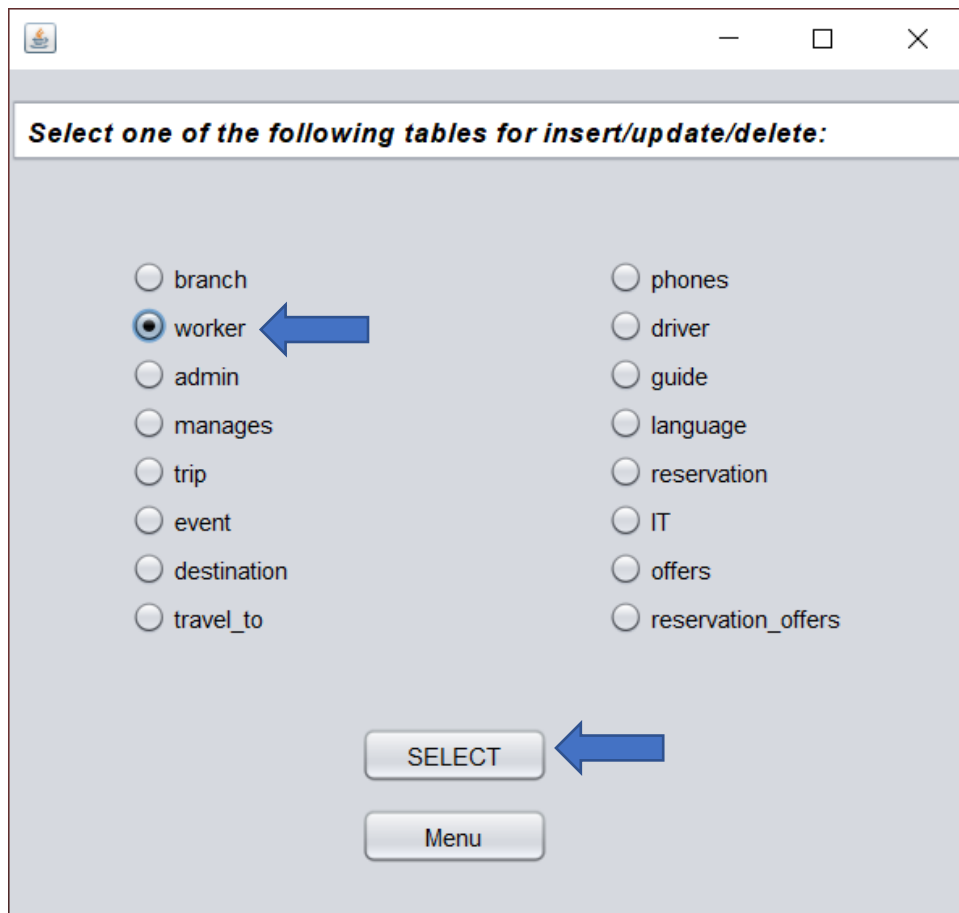
br_code	wrk_lname	wrk_name	wrk_salary
1	Martinez	Creath	1000.0
1	Martinez	Lachance	3000.0
1	Miller	Aarika	2000.0
1	Davis	Nadia	1000.0
1	Jackson	Nadiya	2000.0
2	Garcia	Nadiya	1000.0
2	Rodriguez	Etheline	4000.0
2	Anderson	Nadler	2000.0
2	Rodriguez	Aaren	4000.0
2	Johnson	Lachlan	3000.0
3	Johnson	Lachish	1000.0
3	Wilson	Aaron	2000.0
3	Williams	Nadler	4000.0
3	Jones	Creath	3000.0
3	Hernandez	Lachlan	1000.0

br_code	sumofsalary
1	9000.0
2	14000.0
3	11000.0
4	12000.0
5	13000.0
6	14000.0
7	7000.0
8	10000.0
9	15000.0
10	14000.0

Επιλέγουμε από το μενού την επιλογή “INSERT/UPDATE/DELETE” και πατάμε “SELECT”.



Επιλέγουμε τον πίνακα “worker” και πατάμε “SELECT”.



The screenshot shows a window titled "Select one of the following tables for insert/update/delete:". Inside the window, there are two columns of radio buttons, each followed by a table name. The first column contains: branch, worker, admin, manages, trip, event, destination, and travel_to. The second column contains: phones, driver, guide, language, reservation, IT, offers, and reservation_offers. The "worker" radio button is selected, and a blue arrow points to it. At the bottom of the window, there are two buttons: "SELECT" and "Menu". A blue arrow points to the "SELECT" button.

Select one of the following tables for insert/update/delete:

☐ branch

☒ worker

☐ admin

☐ manages

☐ trip

☐ event

☐ destination

☐ travel_to

☐ phones

☐ driver

☐ guide

☐ language

☐ reservation

☐ IT

☐ offers

☐ reservation_offers

SELECT

Menu

Βάζουμε στα πεδία τα στοιχεία του εργαζόμενου και πατάμε “INSERT”.

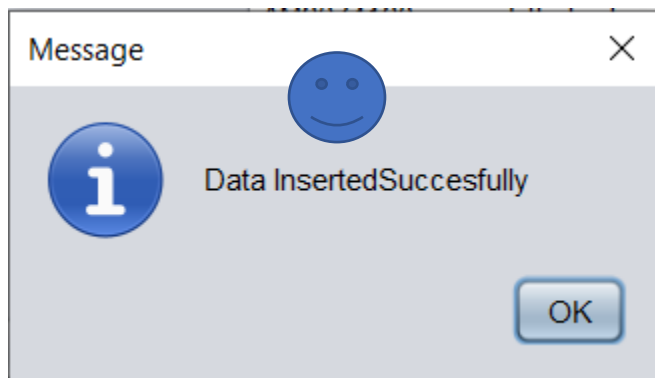
Form fields:

- wrk_AT: AM1231231
- wrk_name: Mitsos
- wrk_lanme: Pitsos
- wrk_salary: 1234.0
- wrk_br_code: 2

Buttons: INSERT, UPDATE, DELETE, Menu

wrk_AT	wrk_name	wrk_lname	wrk_salary	wrk_br_code
AM1321755	Aarika	Garcia	5000.0	10
AM1544854	Lachish	Thompson	3000.0	8
AM1619922	Etheline	Rodriguez	4000.0	2
AM1858528	Creath	Martinez	1000.0	1
AM1893557	Nadiya	Davis	1000.0	9
AM2129858	Creigh	Williams	2000.0	7
AM2512172	Ethelred	Thompson	4000.0	4
AM2674183	Ethelred	Martinez	4000.0	10
AM2761138	Lach	Moore	1000.0	4
AM2844976	Lachish	Williams	4000.0	4
AM2885413	Nadler	Thompson	2000.0	5
AM3128853	Lachance	Anderson	2000.0	8
AM3132639	Nadler	Williams	4000.0	3
AM3656761	Aaren	Wilson	4000.0	5
AM3982193	Aaron	Miller	1000.0	7
AM4171974	Nadiya	Jackson	2000.0	1
AM4194673	Ethelyn	Rodriguez	4000.0	5
AM4832485	Lachlan	Anderson	4000.0	6
AM4843271	Creigh	Martinez	4000.0	6
AM4883967	Nadine	Thompson	1000.0	7
AM5188847	Etheline	Smith	2000.0	6
AM5231946	Aarika	Taylor	4000.0	6
AM5443826	Creath	Jones	3000.0	3
AM5697977	Ethelyn	Thompson	3000.0	5
AM5886366	Nadiya	Garcia	1000.0	2

Βγαίνει μήνυμα επιτυχημένης εισαγωγής.



Πατάμε “MENU” και επιστρέφουμε στο μενού.

The screenshot shows a database application window. On the left, there is a form with input fields for 'wrk_AT', 'wrk_name', 'wrk_lanme', 'wrk_salary', and 'wrk_br_code'. Below these fields are buttons for 'INSERT', 'UPDATE', and 'DELETE'. A 'Menu' button is located below these, with a blue arrow pointing to it. On the right, there is a table with the following data:

wrk_AT	wrk_name	wrk_lname	wrk_salary	wrk_br_code
AM1231231	Mitsos	Pitsos	1234.0	2
AM1321755	Aarika	Garcia	5000.0	10
AM1544854	Lachish	Thompson	3000.0	8
AM1619922	Etheline	Rodriguez	4000.0	2
AM1858528	Creath	Martinez	1000.0	1
AM1893557	Nadiya	Davis	1000.0	9
AM2129858	Creigh	Williams	2000.0	7
AM2512172	Ethelred	Thompson	4000.0	4
AM2674183	Ethelred	Martinez	4000.0	10
AM2761138	Lach	Moore	1000.0	4
AM2844976	Lachish	Williams	4000.0	4
AM2885413	Nadler	Thompson	2000.0	5
AM3128853	Lachance	Anderson	2000.0	8
AM3132639	Nadler	Williams	4000.0	3
AM3656761	Aaren	Wilson	4000.0	5
AM3982193	Aaron	Miller	1000.0	7
AM4171974	Nadiya	Jackson	2000.0	1
AM4194673	Ethelyn	Rodriguez	4000.0	5
AM4832485	Lachlan	Anderson	4000.0	6
AM4843271	Creigh	Martinez	4000.0	6
AM4883967	Nadine	Thompson	1000.0	7
AM5188847	Etheline	Smith	2000.0	6
AM5231946	Aarika	Taylor	4000.0	6
AM5443826	Creath	Jones	3000.0	3
AM5697977	Ethelyn	Thompson	3000.0	5

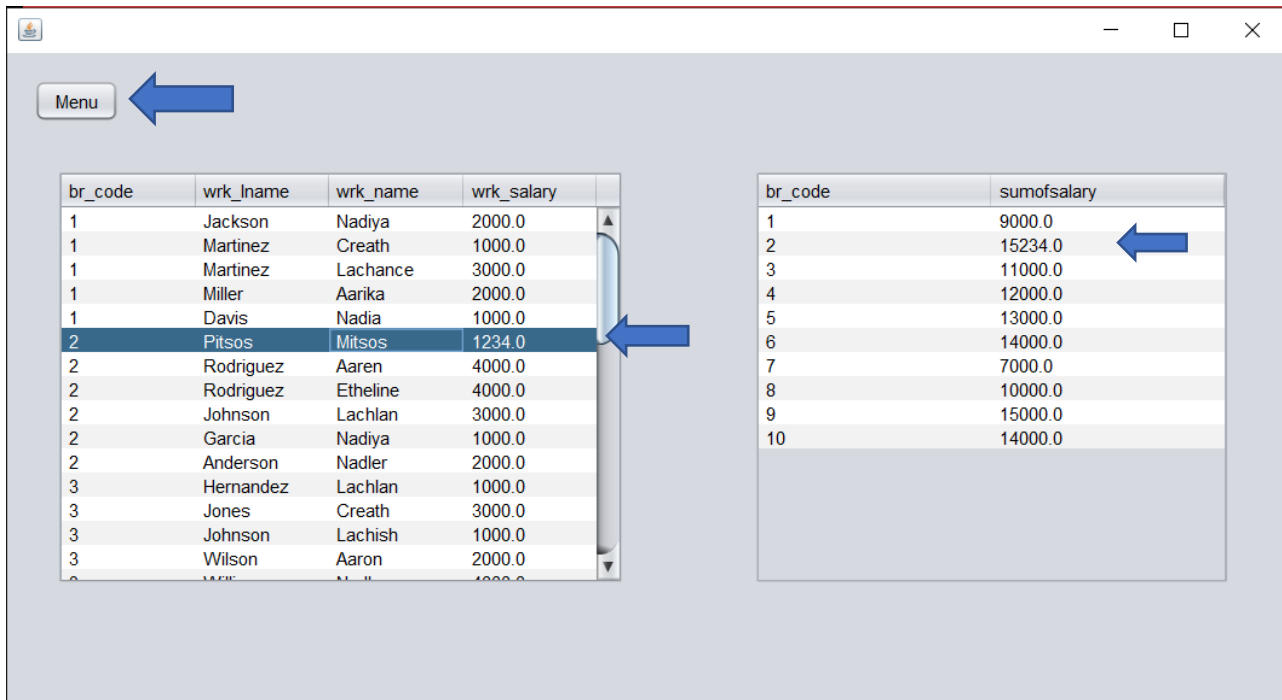
Επιλέγουμε “Branch’s Workers” και πατάμε “SELECT”.

The screenshot shows a menu screen titled "MENU". Below the title, it says "Choose an option to continue". There is a list of options with radio buttons:

- ☐ INSERT/UPDATE/DELETE
- ☐ Search Trip
- ☐ Customer's Reservation
- ☐ Branch's Infos
- ☒ Branch's Workers
- ☐ Show Log
- ☐ Revenue and Cost

To the right of the list, there are two buttons: "SELECT" and "LOG OUT". A blue arrow points to the "SELECT" button.

Βλέπουμε ότι στο υποκατάστημα με br_code=2 το άθροισμα των μιστών τελειώνει σε 234 και άρα επηρεάστηκε από την αλλαγή που κάναμε.

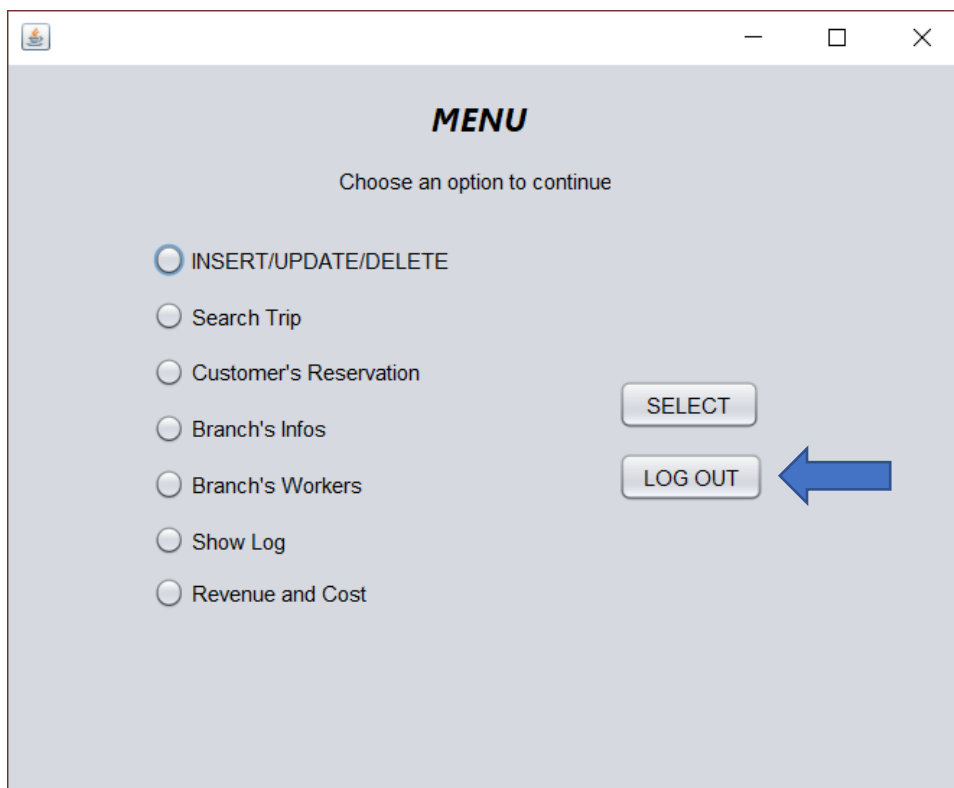


Menu

br_code	wrk_lname	wrk_name	wrk_salary
1	Jackson	Nadiya	2000.0
1	Martinez	Creath	1000.0
1	Martinez	Lachance	3000.0
1	Miller	Aarika	2000.0
1	Davis	Nadia	1000.0
2	Pitsos	Mitsos	1234.0
2	Rodriguez	Aaren	4000.0
2	Rodriguez	Etheline	4000.0
2	Johnson	Lachlan	3000.0
2	Garcia	Nadiya	1000.0
2	Anderson	Nadler	2000.0
3	Hernandez	Lachlan	1000.0
3	Jones	Creath	3000.0
3	Johnson	Lachish	1000.0
3	Wilson	Aaron	2000.0

br_code	sumofsalary
1	9000.0
2	15234.0
3	11000.0
4	12000.0
5	13000.0
6	14000.0
7	7000.0
8	10000.0
9	15000.0
10	14000.0

Πατάμε “MENU”.



MENU

Choose an option to continue

☐ INSERT/UPDATE/DELETE

☐ Search Trip

☐ Customer's Reservation

☐ Branch's Infos

☐ Branch's Workers

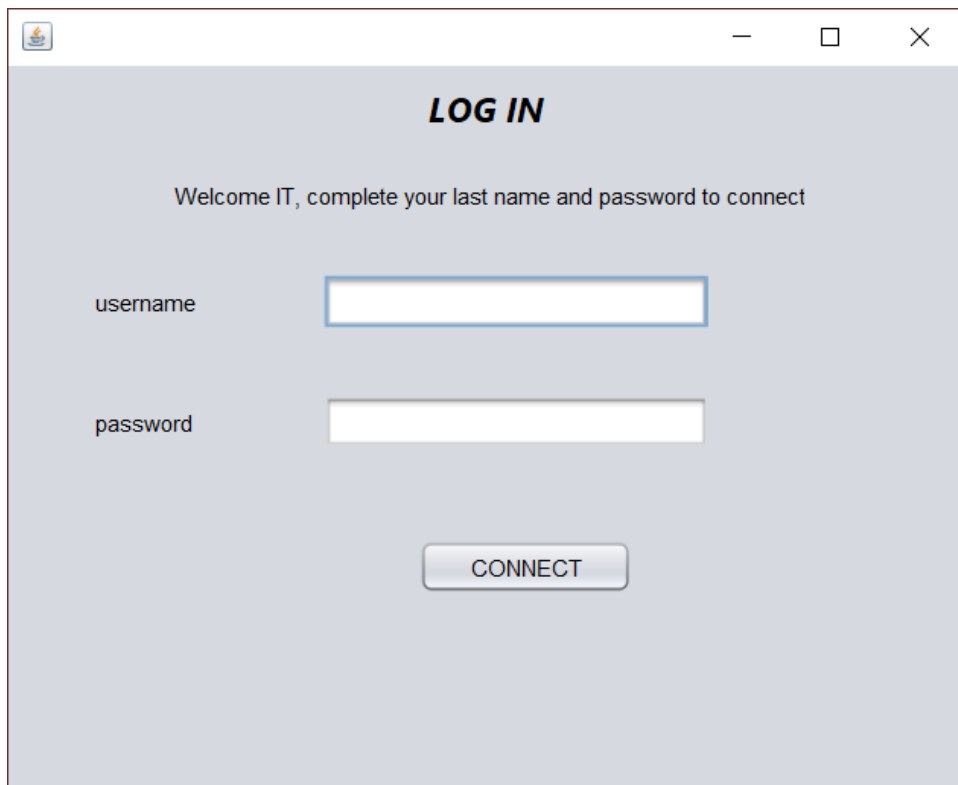
☐ Show Log

☐ Revenue and Cost

SELECT

LOG OUT

Κάνουμε log out πατώντας στο “LOG OUT”.



LOG IN

Welcome IT, complete your last name and password to connect

username

password

5. Extra λειτουργία

Ερώτημα 3.2.3

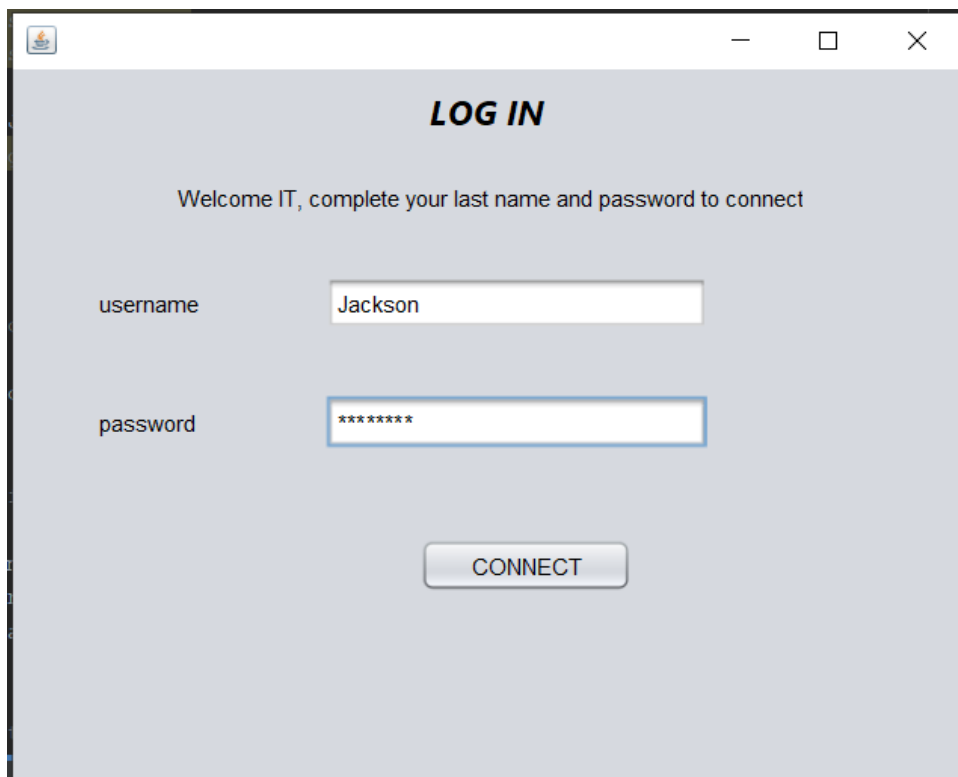
Κλάση RevenueAndCost:

Εκτελούμε κατάλληλο query και τα αποτελέσματα εμφανίζονται στον πίνακα του frame. Στον πίνακα βλέπουμε για όλα τα υποκαταστήματα το συνολικό ποσό εξόδων δηλαδή το άθροισμα των μισθών των υπαλλήλων και το συνολικό ποσό εσόδων δηλαδή κρατήσεις επί κόστος ταξιδιού.

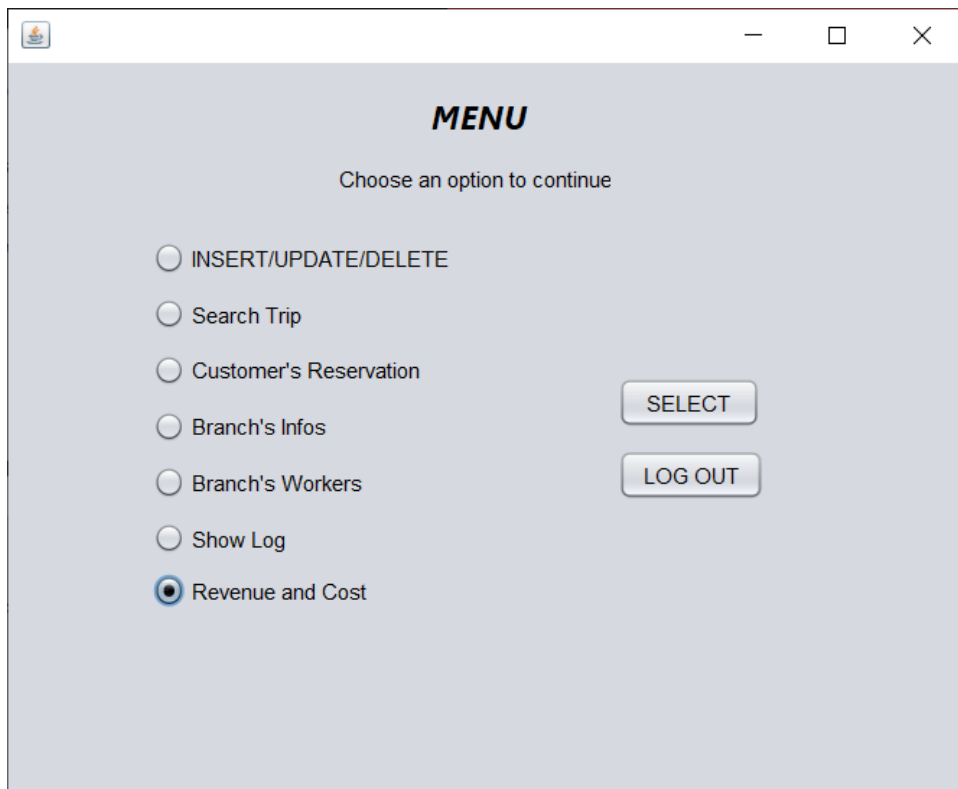
Query που εκτελούμε:

Για να εμφανίσουμε το συνολικό ποσό εξόδων και εξόδων συνδυάζουμε τα queries των κλάσεων BranchInfo και BranchWorker. Για το συνολικό ποσό εξόδων δημιουργούμε προσωρινό πίνακα με τις συνολικές κρατήσεις ανά ταξίδι και το κόστος του ταξιδιού. Το join αυτού του πίνακα με τον άλλο προσωρινό πίνακα που υπολογίζει το σύνολο των εξόδων ανά υποκατάστημα μας δίνει τον τελικό πίνακα που είναι το ζητούμενο.

Κάνουμε log in με έναν IT.



Επιλέγουμε “Revenue and Cost” και πατάμε “SELECT”.



MENU

Choose an option to continue

☐ INSERT/UPDATE/DELETE

☐ Search Trip

☐ Customer's Reservation

☐ Branch's Infos

☐ Branch's Workers

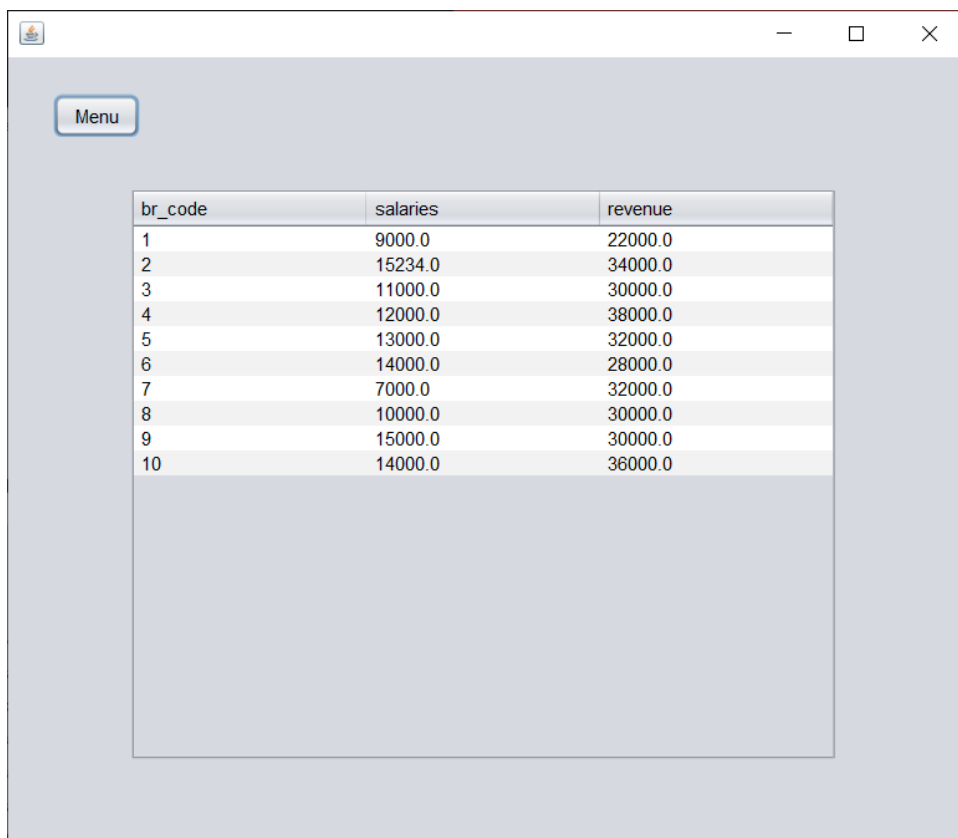
☐ Show Log

☒ Revenue and Cost

SELECT

LOG OUT

Εμφανίζεται στην οθόνη το αποτέλεσμα της εκτέλεσης του query: κωδικός εταιρίας, έξοδα, έσοδα



br_code	salaries	revenue
1	9000.0	22000.0
2	15234.0	34000.0
3	11000.0	30000.0
4	12000.0	38000.0
5	13000.0	32000.0
6	14000.0	28000.0
7	7000.0	32000.0
8	10000.0	30000.0
9	15000.0	30000.0
10	14000.0	36000.0

6. Κώδικες

Εντολές create. Αρχείο 1.create_tables.txt

```
drop database travelagency;
```

```
create database travelagency;
```

```
use travelagency;
```

```
create table branch(
```

```
    br_code int(11) not null AUTO_INCREMENT,
```

```
    br_street varchar(30),
```

```
    br_num int(4),
```

```
    br_city varchar(30),
```

```
    primary key (br_code)
```

```
);
```

```
create table worker(
```

```
    wrk_AT char(10) not null,
```

```
    wrk_name varchar(20),
```

```
    wrk_lname varchar(20),
```

```
    wrk_salary float(7,2),
```

```
    wrk_br_code int(11),
```

```
    primary key (wrk_AT),
```

```
    constraint WorkerBranchCode
```

```
    foreign key(wrk_br_code) REFERENCES branch(br_code)
```

```
    on delete cascade on update cascade
```

```
);
```

```
create table admin(
```

```
    adm_AT char(10) not null,
```

```
    adm_type enum('LOGISTICS','ADMINISTRATIVE','ACCOUNTING'),
```

```
    adm_diploma varchar(200),
```

```
    primary key (adm_AT),
```

```
    constraint AdminWorkerAT
```

```
    foreign key(adm_AT) references worker(wrk_AT)
```

```
    on delete cascade on update cascade
```

```
);
```

```
create table manages(
```

```
    mng_adm_AT char(10) not null,
```

```
    mng_br_code int(11) not null,
```

```

primary key (mng_adm_AT, mng_br_code),

constraint ManagesAdminAT

foreign key(mng_adm_AT) references admin(adm_AT)

on delete cascade on update cascade,

constraint ManagesBranchCode

foreign key(mng_br_code) references branch(br_code)

on delete cascade on update cascade

);

```

```

create table phones(

    ph_br_code int(11) not null,

    ph_number char(10) not null,

    primary key (ph_br_code, ph_number),

    constraint PhonesBranchCode

    foreign key(ph_br_code) references branch(br_code)

    on delete cascade on update cascade

);

```

```

create table driver(

    drv_AT char(10) not null,

    drv_license enum('A','B','C','D'),

    drv_route enum('LOCAL','ABROAD'),

    drv_experience tinyint(4),

    primary key(drv_AT),

    constraint DriverWorkerAT

    foreign key(drv_AT) references worker(wrk_AT)

    on delete cascade on update cascade

);

```

```

create table guide(

    gui_AT char(10) not null,

    gui_cv text,

    primary key(gui_AT),

    constraint GuideWorkerAT

    foreign key(gui_AT) references worker(wrk_AT)

    on delete cascade on update cascade

);

```

```

create table language(

    lng_gui_AT char(10),

    lng_language varchar(30),

    primary key(lng_gui_AT, lng_language),

```

```

constraint LanguageGuideAT

foreign key(lng_gui_AT) references guide(gui_AT)

on delete cascade on update cascade

);

```

```

create table trip(

    tr_id int(11) not null AUTO_INCREMENT,

    tr_departure datetime,

    tr_return datetime,

    tr_maxseats tinyint(4),

    tr_cost float(7,2),

    tr_br_code int(11),

    tr_gui_AT char(10),

    tr_drv_AT char(10),

    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 event(

    ev_tr_id int(11) not null,

    ev_start datetime not null,

    ev_end datetime,

    ev_descr text,

    primary key(ev_tr_id, ev_start),

    constraint EventTripld

    foreign key(ev_tr_id) references trip(tr_id)

    on delete cascade on update cascade

);

```

```

create table destination(

    dst_id int(11) not null AUTO_INCREMENT,

    dst_name varchar(50),

    dst_descr text,

```

```

        dst_rtype enum('LOCAL','ABROAD'),

        dst_language varchar(30),

        dst_location int(11),

        primary key (dst_id),

        constraint DestinationDestinationId

foreign key(dst_location) references destination(dst_id)

on delete cascade on update cascade

);

```

```

create table travel_to(

        to_tr_id int(11) not null,

        to_dst_id int(11) not null,

        to_arrival datetime,

        to_departure datetime,

        primary key(to_tr_id,to_dst_id),

        constraint Travel_toTripId

foreign key(to_tr_id) references trip(tr_id)

on delete cascade on update cascade,

        constraint Travel_toDestinationId

foreign key(to_dst_id) references destination(dst_id)

on delete cascade on update cascade

);

```

```

create table reservation(

        res_tr_id int(11) not null,

        res_seatnum tinyint(4) not null,

        res_name varchar(20),

        res_lname varchar(20),

        res_isadult enum('ADULT','MINOR'),

        primary key(res_tr_id, res_seatnum),

        constraint ReservationTripId

foreign key(res_tr_id) references trip(tr_id)

on delete cascade on update cascade

);

```

```

create table IT(

        IT_AT char(10) not null,

        password varchar(10) default 'password',

        start_date DATE not null,

        end_date DATE,

```



```

        active enum('ACTIVE', 'INACTIVE') default 'INACTIVE',

        primary key (IT_AT),

        constraint ITWorkerAT

        foreign key(IT_AT) references worker(wrk_AT)

        on delete cascade on update cascade

    };

```

```

create table offers(

        off_id int(11) not null auto_increment,

        off_start date,

        off_end date,

        off_cost int(11),

        off_dst_id int(11),

        primary key(off_id),

        constraint OffersDestinationId

        foreign key(off_dst_id) references destination(dst_id)

        on delete cascade on update cascade

);

```

```

create table reservation_offers(

        res_off_id int(11) not null auto_increment,

        res_off_In varchar(250),

        res_off_fn varchar(250),

        res_off_off_id int(11) not null,

        res_off_prepay int(11),

        primary key(res_off_id),

        constraint ResOffersOffersId

        foreign key(res_off_off_id) references offers(off_id)

        on delete cascade on update cascade

);

```

```

create table log(

        num_of_log int (11) not null auto_increment,

        chosentable enum('trip', 'reservation', 'event', 'travel_to', 'destination'),

        action_taken enum('insert', 'delete', 'update') NOT NULL,

        who char(10) DEFAULT 'unknown',

        cur_datetime datetime,

        primary key(num_of_log)

);

```

```
INSERT INTO branch values(NULL,'Josef Stanislaus',9,'Aggsbach');
INSERT INTO branch values(NULL,'Albach-Gasse',45,'Aichbach');
INSERT INTO branch values(NULL,'Römerweg',35,'Aichdorf');
INSERT INTO branch values(NULL,'Sanktstraße',41,'Aifersdorf');
INSERT INTO branch values(NULL,'Antonistraße',19,'Ebensee');
INSERT INTO branch values(NULL,'Augartenweg',42,'Ebenthal');
INSERT INTO branch values(NULL,'Aaxerweg',26,'Ebergassing');
INSERT INTO branch values(NULL,'Bahnhofsplatz',1,'Eberndorf');
INSERT INTO branch values(NULL,'Bahnstraße',41,'Fluess');
INSERT INTO branch values(NULL,'Bankgasse',43,'Flinsdorf');
INSERT INTO worker values('AM9864754','Lachance','Martinez',3000,1);
INSERT INTO worker values('AM6635753','Lachlan','Johnson',3000,2);
INSERT INTO worker values('AM3132639','Nadler','Williams',4000,3);
INSERT INTO worker values('AM2844976','Lachish','Williams',4000,4);
INSERT INTO worker values('AM4194673','Ethelyn','Rodriguez',4000,5);
INSERT INTO worker values('AM4843271','Creigh','Martinez',4000,6);
INSERT INTO worker values('AM6664116','Aaren','Garcia',3000,7);
INSERT INTO worker values('AM1544854','Lachish','Thompson',3000,8);
INSERT INTO worker values('AM1893557','Nadiya','Davis',1000,9);
INSERT INTO worker values('AM9431347','Ethelyn','Anderson',2000,10);
INSERT INTO worker values('AM8782351','Aarika','Miller',2000,1);
INSERT INTO worker values('AM1619922','Etheline','Rodriguez',4000,2);
INSERT INTO worker values('AM8862614','Lachish','Johnson',1000,3);
INSERT INTO worker values('AM2512172','Ethelred','Thompson',4000,4);
INSERT INTO worker values('AM5697977','Ethelyn','Thompson',3000,5);
INSERT INTO worker values('AM5231946','Aarika','Taylor',4000,6);
INSERT INTO worker values('AM2129858','Creigh','Williams',2000,7);
INSERT INTO worker values('AM9772126','Lachish','Anderson',3000,8);
INSERT INTO worker values('AM8997564','Lach','Wilson',4000,9);
INSERT INTO worker values('AM2674183','Ethelred','Martinez',4000,10);
INSERT INTO worker values('AM7326782','Nadia','Davis',1000,1);
INSERT INTO worker values('AM5967597','Aaren','Rodriguez',4000,2);
INSERT INTO worker values('AM5935833','Lachlan','Hernandez',1000,3);
INSERT INTO worker values('AM2761138','Lach','Moore',1000,4);
INSERT INTO worker values('AM3656761','Aaren','Wilson',4000,5);
INSERT INTO worker values('AM5188847','Etheline','Smith',2000,6);
```

```

INSERT INTO worker values('AM4883967','Nadine','Thompson',1000,7);

INSERT INTO worker values('AM3128853','Lachance','Anderson',2000,8);

INSERT INTO worker values('AM6561158','Creath','Brown',5000,9);

INSERT INTO worker values('AM1321755','Aarika','Garcia',5000,10);

INSERT INTO worker values('AM1858528','Creath','Martinez',1000,1);

INSERT INTO worker values('AM5886366','Nadiya','Garcia',1000,2);

INSERT INTO worker values('AM5443826','Creath','Jones',3000,3);

INSERT INTO worker values('AM9714522','Crean','Moore',3000,4);

INSERT INTO worker values('AM2885413','Nadler','Thompson',2000,5);

INSERT INTO worker values('AM4832485','Lachlan','Anderson',4000,6);

INSERT INTO worker values('AM3982193','Aaron','Miller',1000,7);

INSERT INTO worker values('AM6639873','Ethelyn','Anderson',2000,8);

INSERT INTO worker values('AM7927789','Crean','Taylor',5000,9);

INSERT INTO worker values('AM7258361','Aaren','Anderson',3000,10);

INSERT INTO worker values('AM4171974','Nadiya','Jackson',2000,1);

INSERT INTO worker values('AM7993276','Nadler','Anderson',2000,2);

INSERT INTO worker values('AM8453261','Aaron','Wilson',2000,3);

INSERT INTO IT values('AM4171974',DEFAULT,'2015-01-10',NULL,'ACTIVE');

INSERT INTO IT values('AM7993276',DEFAULT,'2015-01-11',NULL,'INACTIVE');

INSERT INTO IT values('AM8453261',DEFAULT,'2015-01-12',NULL,'INACTIVE');

INSERT INTO admin values('AM9864754','ADMINISTRATIVE','Managment');

INSERT INTO admin values('AM6635753','ADMINISTRATIVE','Oikonomika & Management');

INSERT INTO admin values('AM3132639','ADMINISTRATIVE','Managment');

INSERT INTO admin values('AM2844976','ADMINISTRATIVE','Oikonomika & Management');

INSERT INTO admin values('AM4194673','ADMINISTRATIVE','Managment');

INSERT INTO admin values('AM4843271','ADMINISTRATIVE','Managment');

INSERT INTO admin values('AM6664116','ADMINISTRATIVE','Managment');

INSERT INTO admin values('AM1544854','ADMINISTRATIVE','Managment');

INSERT INTO admin values('AM1893557','ADMINISTRATIVE','Oikonomika & Management');

INSERT INTO admin values('AM9431347','ADMINISTRATIVE','Oikonomika & Management');

INSERT INTO admin values('AM8782351','ACCOUNTING','Oikonomika');

INSERT INTO admin values('AM1619922','ACCOUNTING','Oikonomika');

INSERT INTO admin values('AM8862614','ACCOUNTING','Oikonomika & Management');

INSERT INTO admin values('AM2512172','LOGISTICS','Oikonomika & Management');

INSERT INTO admin values('AM5697977','ACCOUNTING','Oikonomika & Management');

INSERT INTO admin values('AM5231946','ACCOUNTING','Oikonomika');

INSERT INTO admin values('AM2129858','LOGISTICS','Oikonomika');

INSERT INTO admin values('AM9772126','ACCOUNTING','Oikonomika & Management');

INSERT INTO admin values('AM8997564','LOGISTICS','Oikonomika');

```

```

INSERT INTO admin values('AM2674183','LOGISTICS','Oikonomika');

INSERT INTO manages values('AM9864754',0);

INSERT INTO manages values('AM6635753',1);

INSERT INTO manages values('AM3132639',2);

INSERT INTO manages values('AM2844976',3);

INSERT INTO manages values('AM4194673',4);

INSERT INTO manages values('AM4843271',5);

INSERT INTO manages values('AM6664116',6);

INSERT INTO manages values('AM1544854',7);

INSERT INTO manages values('AM1893557',8);

INSERT INTO manages values('AM9431347',9);

INSERT INTO phones values(1,2610683568);

INSERT INTO phones values(2,2610527185);

INSERT INTO phones values(3,2610839889);

INSERT INTO phones values(4,2610225565);

INSERT INTO phones values(5,2610781263);

INSERT INTO phones values(6,2610667127);

INSERT INTO phones values(7,2610489977);

INSERT INTO phones values(8,2610292827);

INSERT INTO phones values(9,2610868322);

INSERT INTO phones values(10,2610616725);

INSERT INTO phones values(1,2610423743);

INSERT INTO phones values(2,2610577136);

INSERT INTO phones values(3,2610737169);

INSERT INTO driver values('AM7326782','B','ABROAD',19);

INSERT INTO driver values('AM5967597','C','LOCAL',53);

INSERT INTO driver values('AM5935833','B','LOCAL',100);

INSERT INTO driver values('AM2761138','D','ABROAD',67);

INSERT INTO driver values('AM3656761','C','ABROAD',73);

INSERT INTO driver values('AM5188847','D','LOCAL',19);

INSERT INTO driver values('AM4883967','A','LOCAL',10);

INSERT INTO driver values('AM3128853','A','ABROAD',68);

INSERT INTO driver values('AM6561158','A','LOCAL',46);

INSERT INTO driver values('AM1321755','A','ABROAD',39);

INSERT INTO guide values('AM1858528','Guide 1 CV');

INSERT INTO guide values('AM5886366','Guide 2 CV');

INSERT INTO guide values('AM5443826','Guide 3 CV');

INSERT INTO guide values('AM9714522','Guide 4 CV');

INSERT INTO guide values('AM2885413','Guide 5 CV');

```

```

INSERT INTO guide values('AM4832485','Guide 6 CV');

INSERT INTO guide values('AM3982193','Guide 7 CV');

INSERT INTO guide values('AM6639873','Guide 8 CV');

INSERT INTO guide values('AM7927789','Guide 9 CV');

INSERT INTO guide values('AM7258361','Guide 10 CV');

INSERT INTO language values('AM1858528','Albanian');

INSERT INTO language values('AM5886366','German');

INSERT INTO language values('AM5443826','Hindi');

INSERT INTO language values('AM9714522','Russian');

INSERT INTO language values('AM2885413','Serbian');

INSERT INTO language values('AM4832485','Chinese');

INSERT INTO language values('AM3982193','Greek');

INSERT INTO language values('AM6639873','Portuguese');

INSERT INTO language values('AM7927789','French');

INSERT INTO language values('AM7258361','English');

INSERT INTO language values('AM1858528','Sweedish');

INSERT INTO language values('AM5886366','Polish');

INSERT INTO language values('AM5443826','Albanian');

INSERT INTO language values('AM9714522','German');

INSERT INTO language values('AM2885413','Hindi');

INSERT INTO trip values(NULL,'2022-11-10 09:00:00','2022-12-13 12:00:00',32,600,1,'AM1858528','AM7326782');

INSERT INTO trip values(NULL,'2022-11-11 09:00:00','2022-12-14 12:00:00',25,900,2,'AM5886366','AM5967597');

INSERT INTO trip values(NULL,'2022-11-12 09:00:00','2022-12-15 12:00:00',26,900,3,'AM5443826','AM5935833');

INSERT INTO trip values(NULL,'2022-11-13 09:00:00','2022-12-16 12:00:00',32,900,4,'AM9714522','AM2761138');

INSERT INTO trip values(NULL,'2022-11-14 09:00:00','2022-12-17 12:00:00',32,900,5,'AM2885413','AM3656761');

INSERT INTO trip values(NULL,'2022-11-15 09:00:00','2022-12-18 12:00:00',28,800,6,'AM4832485','AM5188847');

INSERT INTO trip values(NULL,'2022-11-16 09:00:00','2022-12-19 12:00:00',32,600,7,'AM3982193','AM4883967');

INSERT INTO trip values(NULL,'2022-11-17 09:00:00','2022-12-20 12:00:00',30,600,8,'AM6639873','AM3128853');

INSERT INTO trip values(NULL,'2022-11-18 09:00:00','2022-12-21 12:00:00',33,800,9,'AM7927789','AM6561158');

INSERT INTO trip values(NULL,'2022-11-19 09:00:00','2022-12-22 12:00:00',34,900,10,'AM7258361','AM1321755');

INSERT INTO trip values(NULL,'2022-11-10 09:00:00','2022-12-13 12:00:00',26,500,1,'AM1858528','AM7326782');

INSERT INTO trip values(NULL,'2022-11-11 09:00:00','2022-12-14 12:00:00',29,800,2,'AM5886366','AM5967597');

INSERT INTO trip values(NULL,'2022-11-12 09:00:00','2022-12-15 12:00:00',27,600,3,'AM5443826','AM5935833');

INSERT INTO trip values(NULL,'2022-11-13 09:00:00','2022-12-16 12:00:00',27,1000,4,'AM9714522','AM2761138');

INSERT INTO trip values(NULL,'2022-11-14 09:00:00','2022-12-17 12:00:00',26,700,5,'AM2885413','AM3656761');

INSERT INTO trip values(NULL,'2022-11-15 09:00:00','2022-12-18 12:00:00',27,600,6,'AM4832485','AM5188847');

INSERT INTO trip values(NULL,'2022-11-16 09:00:00','2022-12-19 12:00:00',30,1000,7,'AM3982193','AM4883967');

INSERT INTO trip values(NULL,'2022-11-17 09:00:00','2022-12-20 12:00:00',25,900,8,'AM6639873','AM3128853');

INSERT INTO trip values(NULL,'2022-11-18 09:00:00','2022-12-21 12:00:00',28,700,9,'AM7927789','AM6561158');

```

```

INSERT INTO trip values(NULL,'2022-11-19 09:00:00','2022-12-22 12:00:00',30,900,10,'AM7258361','AM1321755');

INSERT INTO event values(1,'2022-11-10 09:00:00','2022-12-13 12:00:00','Event 1 Description!');

INSERT INTO event values(2,'2022-11-11 09:00:00','2022-12-14 12:00:00','Event 2 Description!');

INSERT INTO event values(3,'2022-11-12 09:00:00','2022-12-15 12:00:00','Event 3 Description!');

INSERT INTO event values(4,'2022-11-13 09:00:00','2022-12-16 12:00:00','Event 4 Description!');

INSERT INTO event values(5,'2022-11-14 09:00:00','2022-12-17 12:00:00','Event 5 Description!');

INSERT INTO event values(6,'2022-11-15 09:00:00','2022-12-18 12:00:00','Event 6 Description!');

INSERT INTO event values(7,'2022-11-16 09:00:00','2022-12-19 12:00:00','Event 7 Description!');

INSERT INTO event values(8,'2022-11-17 09:00:00','2022-12-20 12:00:00','Event 8 Description!');

INSERT INTO event values(9,'2022-11-18 09:00:00','2022-12-21 12:00:00','Event 9 Description!');

INSERT INTO event values(10,'2022-11-19 09:00:00','2022-12-22 12:00:00','Event 10 Description!');

INSERT INTO event values(11,'2022-11-10 09:00:00','2022-12-13 12:00:00','Event 11 Description!');

INSERT INTO event values(12,'2022-11-11 09:00:00','2022-12-14 12:00:00','Event 12 Description!');

INSERT INTO event values(13,'2022-11-12 09:00:00','2022-12-15 12:00:00','Event 13 Description!');

INSERT INTO event values(14,'2022-11-13 09:00:00','2022-12-16 12:00:00','Event 14 Description!');

INSERT INTO event values(15,'2022-11-14 09:00:00','2022-12-17 12:00:00','Event 15 Description!');

INSERT INTO event values(16,'2022-11-15 09:00:00','2022-12-18 12:00:00','Event 16 Description!');

INSERT INTO event values(17,'2022-11-16 09:00:00','2022-12-19 12:00:00','Event 17 Description!');

INSERT INTO event values(18,'2022-11-17 09:00:00','2022-12-20 12:00:00','Event 18 Description!');

INSERT INTO event values(19,'2022-11-18 09:00:00','2022-12-21 12:00:00','Event 19 Description!');

INSERT INTO event values(20,'2022-11-19 09:00:00','2022-12-22 12:00:00','Event 20 Description!');

INSERT INTO destination values(NULL,'Albania','Cool Place!','ABROAD','Albanian',NULL);

INSERT INTO destination values(NULL,'Germany','Cool Place!','LOCAL','German',NULL);

INSERT INTO destination values(NULL,'India','Cool Place!','LOCAL','Hindi',NULL);

INSERT INTO destination values(NULL,'Russia','Cool Place!','ABROAD','Russian',NULL);

INSERT INTO destination values(NULL,'Serbia','Cool Place!','ABROAD','Serbian',NULL);

INSERT INTO destination values(NULL,'China','Cool Place!','LOCAL','Chinese',NULL);

INSERT INTO destination values(NULL,'Greece','Cool Place!','LOCAL','Greek',NULL);

INSERT INTO destination values(NULL,'Portugal','Cool Place!','ABROAD','Portuguese',NULL);

INSERT INTO destination values(NULL,'France','Cool Place!','LOCAL','French',NULL);

INSERT INTO destination values(NULL,'UK','Cool Place!','ABROAD','English',NULL);

INSERT INTO destination values(NULL,'Albania','Cool Place!','ABROAD','Albanian',NULL);

INSERT INTO destination values(NULL,'Germany','Cool Place!','LOCAL','German',NULL);

INSERT INTO destination values(NULL,'India','Cool Place!','LOCAL','Hindi',NULL);

INSERT INTO destination values(NULL,'Russia','Cool Place!','ABROAD','Russian',NULL);

INSERT INTO destination values(NULL,'Serbia','Cool Place!','ABROAD','Serbian',NULL);

INSERT INTO destination values(NULL,'China','Cool Place!','LOCAL','Chinese',NULL);

INSERT INTO destination values(NULL,'Greece','Cool Place!','LOCAL','Greek',NULL);

INSERT INTO destination values(NULL,'Portugal','Cool Place!','ABROAD','Portuguese',NULL);

```

```

INSERT INTO destination values(NULL,'France','Cool Place!','LOCAL','French',NULL);

INSERT INTO destination values(NULL,'UK','Cool Place!','ABROAD','English',NULL);

INSERT INTO travel_to values(1,1,'2022-11-10 09:00:00','2022-12-13 12:00:00');

INSERT INTO travel_to values(2,2,'2022-11-11 09:00:00','2022-12-14 12:00:00');

INSERT INTO travel_to values(3,3,'2022-11-12 09:00:00','2022-12-15 12:00:00');

INSERT INTO travel_to values(4,4,'2022-11-13 09:00:00','2022-12-16 12:00:00');

INSERT INTO travel_to values(5,5,'2022-11-14 09:00:00','2022-12-17 12:00:00');

INSERT INTO travel_to values(6,6,'2022-11-15 09:00:00','2022-12-18 12:00:00');

INSERT INTO travel_to values(7,7,'2022-11-16 09:00:00','2022-12-19 12:00:00');

INSERT INTO travel_to values(8,8,'2022-11-17 09:00:00','2022-12-20 12:00:00');

INSERT INTO travel_to values(9,9,'2022-11-18 09:00:00','2022-12-21 12:00:00');

INSERT INTO travel_to values(10,10,'2022-11-19 09:00:00','2022-12-22 12:00:00');

INSERT INTO travel_to values(11,11,'2022-11-10 09:00:00','2022-12-13 12:00:00');

INSERT INTO travel_to values(12,12,'2022-11-11 09:00:00','2022-12-14 12:00:00');

INSERT INTO travel_to values(13,13,'2022-11-12 09:00:00','2022-12-15 12:00:00');

INSERT INTO travel_to values(14,14,'2022-11-13 09:00:00','2022-12-16 12:00:00');

INSERT INTO travel_to values(15,15,'2022-11-14 09:00:00','2022-12-17 12:00:00');

INSERT INTO travel_to values(16,16,'2022-11-15 09:00:00','2022-12-18 12:00:00');

INSERT INTO travel_to values(17,17,'2022-11-16 09:00:00','2022-12-19 12:00:00');

INSERT INTO travel_to values(18,18,'2022-11-17 09:00:00','2022-12-20 12:00:00');

INSERT INTO travel_to values(19,19,'2022-11-18 09:00:00','2022-12-21 12:00:00');

INSERT INTO travel_to values(20,20,'2022-11-19 09:00:00','2022-12-22 12:00:00');

INSERT INTO reservation values(1,1,'Kellie','Owens','MINOR');

INSERT INTO reservation values(2,1,'Ricard','Gomez','ADULT');

INSERT INTO reservation values(3,1,'Eachelle','Sanders','ADULT');

INSERT INTO reservation values(4,1,'Amberly','Morales','MINOR');

INSERT INTO reservation values(5,1,'Suzanne','Shaw','MINOR');

INSERT INTO reservation values(6,1,'Georgetta','Lopez','MINOR');

INSERT INTO reservation values(7,1,'Justinian','West','MINOR');

INSERT INTO reservation values(8,1,'Dustie','Holmes','ADULT');

INSERT INTO reservation values(9,1,'Oralia','Marshall','ADULT');

INSERT INTO reservation values(10,1,'Janelle','Scott','MINOR');

INSERT INTO reservation values(11,1,'Spanos','Gray','MINOR');

INSERT INTO reservation values(12,1,'Moth','Powell','MINOR');

INSERT INTO reservation values(13,1,'Daniela','Alvarez','MINOR');

INSERT INTO reservation values(14,1,'Boiney','Green','ADULT');

INSERT INTO reservation values(15,1,'Ammamaria','Bailey','ADULT');

INSERT INTO reservation values(16,1,'Castor','Hunter','MINOR');

INSERT INTO reservation values(17,1,'Corell','Porter','MINOR');

```

```

INSERT INTO reservation values(18,1,'Yoho','Nelson','ADULT');

INSERT INTO reservation values(19,1,'Rosetta','Mcdonald','MINOR');

INSERT INTO reservation values(20,1,'Faus','Johnson','ADULT');

INSERT INTO reservation values(1,1,'Moth','Cruz','ADULT');

INSERT INTO reservation values(2,1,'Zela','Meyer','MINOR');

INSERT INTO reservation values(3,1,'Denie','Rogers','MINOR');

INSERT INTO reservation values(4,1,'Citron','Reed','ADULT');

INSERT INTO reservation values(5,1,'Hildegarde','Murray','MINOR');

INSERT INTO reservation values(6,1,'Babara','Nelson','ADULT');

INSERT INTO reservation values(7,1,'Devland','Cole','ADULT');

INSERT INTO reservation values(8,1,'Pettifer','Ruiz','ADULT');

INSERT INTO reservation values(9,1,'Keeley','Snyder','MINOR');

INSERT INTO reservation values(10,1,'Nellie','Simpson','MINOR');

INSERT INTO reservation values(11,1,'Averyl','Nelson','MINOR');

INSERT INTO reservation values(12,1,'Ruttger','Bryant','ADULT');

INSERT INTO reservation values(13,1,'Robbins','Rogers','ADULT');

INSERT INTO reservation values(14,1,'Wernher','Coleman','ADULT');

INSERT INTO reservation values(15,1,'Yoho','Evans','ADULT');

INSERT INTO reservation values(16,1,'September','Hughes','MINOR');

INSERT INTO reservation values(17,1,'Clyve','Jimenez','MINOR');

INSERT INTO reservation values(18,1,'Perrie','Ford','MINOR');

INSERT INTO reservation values(19,1,'Nesta','Robertson','MINOR');

INSERT INTO reservation values(20,1,'Hugues','Fisher','ADULT');

INSERT INTO reservation values(1,1,'Orville','Morris','ADULT');

INSERT INTO reservation values(2,1,'Justinian','Gonzalez','ADULT');

INSERT INTO reservation values(3,1,'Hankins','Rodriguez','ADULT');

INSERT INTO reservation values(4,1,'Eisenstark','Dixon','MINOR');

INSERT INTO reservation values(5,1,'Kennett','Long','MINOR');

INSERT INTO reservation values(6,1,'Edina','Murphy','MINOR');

INSERT INTO reservation values(7,1,'LaMori','Johnson','ADULT');

INSERT INTO reservation values(8,1,'Tanney','Edwards','MINOR');

INSERT INTO reservation values(9,1,'Colon','Campbell','MINOR');

INSERT INTO reservation values(10,1,'Conias','Jackson','ADULT');

INSERT INTO reservation values(11,1,'Renelle','Henderson','MINOR');

INSERT INTO reservation values(12,1,'Brookhouse','Rivera','MINOR');

INSERT INTO reservation values(13,1,'Cavallaro','King','MINOR');

INSERT INTO reservation values(14,1,'Linker','Howard','MINOR');

INSERT INTO reservation values(15,1,'Woodrow','Holmes','MINOR');

INSERT INTO reservation values(16,1,'Ran','Miller','MINOR');

```



```

INSERT INTO reservation values(17,1,'Dream','Perry','MINOR');

INSERT INTO reservation values(18,1,'Citron','Griffin','MINOR');

INSERT INTO reservation values(19,1,'Valdis','Hamilton','MINOR');

INSERT INTO reservation values(20,1,'Yoho','Cook','ADULT');

INSERT INTO reservation values(1,1,'Dolphin','White','ADULT');

INSERT INTO reservation values(2,1,'Fenner','Flores','ADULT');

INSERT INTO reservation values(3,1,'Ron','Porter','ADULT');

INSERT INTO reservation values(4,1,'Gamali','Robinson','ADULT');

INSERT INTO reservation values(5,1,'Puritan','Burns','ADULT');

INSERT INTO reservation values(6,1,'Vitoria','Allen','MINOR');

INSERT INTO reservation values(7,1,'Sholes','Alexander','ADULT');

INSERT INTO reservation values(8,1,'Rocky','King','ADULT');

INSERT INTO reservation values(9,1,'Pollux','Stone','MINOR');

INSERT INTO reservation values(10,1,'Eryn','Morgan','MINOR');

INSERT INTO reservation values(11,1,'Micco','Green','MINOR');

INSERT INTO reservation values(12,1,'Hersch','Sullivan','MINOR');

INSERT INTO reservation values(13,1,'Ammamaria','Dixon','ADULT');

INSERT INTO reservation values(14,1,'Maurene','Castillo','ADULT');

INSERT INTO reservation values(15,1,'Capriola','Griffin','MINOR');

INSERT INTO reservation values(16,1,'Mellman','Taylor','ADULT');

INSERT INTO reservation values(17,1,'Shanleigh','King','ADULT');

INSERT INTO reservation values(18,1,'Spanos','Martin','ADULT');

INSERT INTO reservation values(19,1,'Zimmermann','Lee','MINOR');

INSERT INTO reservation values(20,1,'Micco','Perry','ADULT');

INSERT INTO reservation values(1,1,'Adalard','Hamilton','ADULT');

INSERT INTO reservation values(2,1,'Daryn','Bell','MINOR');

INSERT INTO reservation values(3,1,'Leander','Rogers','MINOR');

INSERT INTO reservation values(4,1,'Theodosia','Long','MINOR');

INSERT INTO reservation values(5,1,'Kimber','Mitchell','MINOR');

INSERT INTO reservation values(6,1,'Johnson','Webb','MINOR');

INSERT INTO reservation values(7,1,'Ninetta','Russell','MINOR');

INSERT INTO reservation values(8,1,'Darda','Perry','MINOR');

INSERT INTO reservation values(9,1,'Cadmann','Clark','MINOR');

INSERT INTO reservation values(10,1,'Meras','Griffin','MINOR');

INSERT INTO reservation values(11,1,'Lorianna','Cox','MINOR');

INSERT INTO reservation values(12,1,'Evelyn','Patterson','ADULT');

INSERT INTO reservation values(13,1,'Andy','Lopez','ADULT');

INSERT INTO reservation values(14,1,'Samantha','Harris','ADULT');

INSERT INTO reservation values(15,1,'Oberheim','Young','MINOR');

```

```

INSERT INTO reservation values(16,1,'Campagna','Robertson','ADULT');

INSERT INTO reservation values(17,1,'Devland','Nguyen','ADULT');

INSERT INTO reservation values(18,1,'Ferullo','Campbell','ADULT');

INSERT INTO reservation values(19,1,'Alfie','Carter','MINOR');

INSERT INTO reservation values(20,1,'Claudina','Ford','ADULT');

INSERT INTO reservation values(1,1,'Apollus','Jordan','ADULT');

INSERT INTO reservation values(2,1,'Wilber','Garcia','MINOR');

INSERT INTO reservation values(3,1,'Sibella','Graham','ADULT');

INSERT INTO reservation values(4,1,'Burkitt','Taylor','ADULT');

INSERT INTO reservation values(5,1,'Pain','Robinson','MINOR');

INSERT INTO reservation values(6,1,'Ethan','Wood','MINOR');

INSERT INTO reservation values(7,1,'Ankney','Lee','ADULT');

INSERT INTO reservation values(8,1,'Silvester','Hayes','MINOR');

INSERT INTO reservation values(9,1,'Jerrol','Foster','MINOR');

INSERT INTO reservation values(10,1,'Thomasin','Bell','MINOR');

INSERT INTO reservation values(11,1,'Ridglee','Rodriguez','ADULT');

INSERT INTO reservation values(12,1,'Edina','Johnson','MINOR');

INSERT INTO reservation values(13,1,'Mafalda','Cook','MINOR');

INSERT INTO reservation values(14,1,'Vitoria','Wood','ADULT');

INSERT INTO reservation values(15,1,'Tu','Wells','MINOR');

INSERT INTO reservation values(16,1,'Secor','Davis','MINOR');

INSERT INTO reservation values(17,1,'Baten','Wells','ADULT');

INSERT INTO reservation values(18,1,'Andy','Romero','MINOR');

INSERT INTO reservation values(19,1,'Millburn','Martin','ADULT');

INSERT INTO reservation values(20,1,'Micco','Bailey','ADULT');

INSERT INTO reservation values(1,1,'Walter','West','MINOR');

INSERT INTO reservation values(2,1,'Deedee','Nelson','MINOR');

INSERT INTO reservation values(3,1,'Ulda','Tucker','MINOR');

INSERT INTO reservation values(4,1,'Samantha','Meyer','ADULT');

INSERT INTO reservation values(5,1,'Koser','Brooks','ADULT');

INSERT INTO reservation values(6,1,'Hadik','Howard','MINOR');

INSERT INTO reservation values(7,1,'Ron','Gutierrez','MINOR');

INSERT INTO reservation values(8,1,'Torey','Alexander','MINOR');

INSERT INTO reservation values(9,1,'Orlena','Campbell','MINOR');

INSERT INTO reservation values(10,1,'Parry','Gutierrez','MINOR');

INSERT INTO reservation values(11,1,'Bernat','Cook','MINOR');

INSERT INTO reservation values(12,1,'Tallu','Owens','ADULT');

INSERT INTO reservation values(13,1,'Sibella','Gonzalez','ADULT');

INSERT INTO reservation values(14,1,'Thomasin','Vasquez','ADULT');

```

```

INSERT INTO reservation values(15,1,'Walter','Hunter','ADULT');

INSERT INTO reservation values(16,1,'De','Clark','ADULT');

INSERT INTO reservation values(17,1,'Zimmermann','King','MINOR');

INSERT INTO reservation values(18,1,'Williamson','Gomez','MINOR');

INSERT INTO reservation values(19,1,'Yun','Harrison','ADULT');

INSERT INTO reservation values(20,1,'Redmer','Wallace','MINOR');

INSERT INTO reservation values(1,1,'Marylin','King','MINOR');

INSERT INTO reservation values(2,1,'Bonny','Ruiz','MINOR');

INSERT INTO reservation values(3,1,'Guinna','Fisher','MINOR');

INSERT INTO reservation values(4,1,'Elish','Nguyen','MINOR');

INSERT INTO reservation values(5,1,'Gonroff','Rivera','ADULT');

INSERT INTO reservation values(6,1,'Sholes','Mason','ADULT');

INSERT INTO reservation values(7,1,'McCormick','Lopez','ADULT');

INSERT INTO reservation values(8,1,'Citron','Porter','MINOR');

INSERT INTO reservation values(9,1,'Abisha','Butler','ADULT');

INSERT INTO reservation values(10,1,'Gina','Flores','ADULT');

INSERT INTO reservation values(11,1,'Utham','Thomas','MINOR');

INSERT INTO reservation values(12,1,'Maye','Castillo','MINOR');

INSERT INTO reservation values(13,1,'Theodosia','Rodriguez','ADULT');

INSERT INTO reservation values(14,1,'Crawley','Allen','MINOR');

INSERT INTO reservation values(15,1,'Glick','Collins','ADULT');

INSERT INTO reservation values(16,1,'Silvester','Robinson','MINOR');

INSERT INTO reservation values(17,1,'Alithia','Vasquez','MINOR');

INSERT INTO reservation values(18,1,'Zela','Owens','ADULT');

INSERT INTO reservation values(19,1,'Grosvenor','Stewart','ADULT');

INSERT INTO reservation values(20,1,'Gaspar','Ross','ADULT');

INSERT INTO reservation values(1,1,'Oler','Walker','ADULT');

INSERT INTO reservation values(2,1,'Albertina','West','ADULT');

INSERT INTO reservation values(3,1,'Eisenstark','Shaw','MINOR');

INSERT INTO reservation values(4,1,'Cadmann','Morales','ADULT');

INSERT INTO reservation values(5,1,'Lancot','Lopez','MINOR');

INSERT INTO reservation values(6,1,'Apollus','Diaz','MINOR');

INSERT INTO reservation values(7,1,'Beal','Collins','ADULT');

INSERT INTO reservation values(8,1,'Deedee','Ellis','ADULT');

INSERT INTO reservation values(9,1,'Lou','Holmes','MINOR');

INSERT INTO reservation values(10,1,'Hardman','Holmes','ADULT');

INSERT INTO reservation values(11,1,'Averyl','Jackson','ADULT');

INSERT INTO reservation values(12,1,'Kennett','Martin','MINOR');

INSERT INTO reservation values(13,1,'Baten','Vasquez','MINOR');

```

```

INSERT INTO reservation values(14,1,'Delmar','Brown','ADULT');

INSERT INTO reservation values(15,1,'Shaylyn','Ramos','ADULT');

INSERT INTO reservation values(16,1,'Boiney','Carter','ADULT');

INSERT INTO reservation values(17,1,'Chaunce','Gonzalez','ADULT');

INSERT INTO reservation values(18,1,'Clere','Wells','MINOR');

INSERT INTO reservation values(19,1,'Brookhouse','Bennett','ADULT');

INSERT INTO reservation values(20,1,'Campagna','Stevens','MINOR');

INSERT INTO reservation values(1,1,'Panter','Hayes','MINOR');

INSERT INTO reservation values(2,1,'Hepsiba','Wagner','ADULT');

INSERT INTO reservation values(3,1,'Henden','Gray','ADULT');

INSERT INTO reservation values(4,1,'Merrick','Mcdonald','ADULT');

INSERT INTO reservation values(5,1,'Cherlyn','Garcia','MINOR');

INSERT INTO reservation values(6,1,'Cyb','Harrison','ADULT');

INSERT INTO reservation values(7,1,'Citron','Gutierrez','ADULT');

INSERT INTO reservation values(8,1,'Warring','Mendoza','MINOR');

INSERT INTO reservation values(9,1,'Tish','James','MINOR');

INSERT INTO reservation values(10,1,'Corell','Johnson','MINOR');

INSERT INTO reservation values(11,1,'Plath','Butler','MINOR');

INSERT INTO reservation values(12,1,'Edyth','Alvarez','ADULT');

INSERT INTO reservation values(13,1,'Aulea','Palmer','MINOR');

INSERT INTO reservation values(14,1,'Vincents','Bennett','ADULT');

INSERT INTO reservation values(15,1,'Squier','Bailey','ADULT');

INSERT INTO reservation values(16,1,'Renelle','Rodriguez','ADULT');

INSERT INTO reservation values(17,1,'Philbo','Simmons','MINOR');

INSERT INTO reservation values(18,1,'Ancel','Smith','MINOR');

INSERT INTO reservation values(19,1,'Colon','Black','ADULT');

INSERT INTO reservation values(20,1,'Rumpf','Palmer','ADULT');

insert into offers values(NULL, '2023-11-18', '2023-11-28', 800, 2); insert into offers values(NULL, '2023-06-18', '2023-06-25', 500, 6); insert into offers values(NULL, '2023-03-18', '2023-03-23', 300, 9);

```

Εντολές αρχείων *.java

```
package bd;

import java.sql.*;
import java.util.ArrayList;
import javax.swing.*;
import javax.swing.table.*;

public class Admin extends javax.swing.JFrame {

    public Admin() {
        initComponents();
        this.setLocationRelativeTo(null);
        DisplayTable();
    }

    public ArrayList<SetterAdmin> tableList(){

        ArrayList<SetterAdmin> tableList = new ArrayList<>();
        Connection con = (new GetConnection()).getConnection();
        String query = "select * from admin ";
        Statement st;
        ResultSet rs;

        try{
            st = con.createStatement();
            rs = st.executeQuery(query);
            SetterAdmin setter;
            while(rs.next()){
                setter = new SetterAdmin(rs.getString("adm_AT"), rs.getString("adm_type"), rs.getString("adm_diploma"));
                tableList.add(setter);
            }
        }catch(Exception ex){
            ex.printStackTrace();
        }
    }
}
```

```

return tableList;
}

private void DisplayTable(){
    ArrayList<SetterAdmin> list = tableList();
    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();
    Object [] row = new Object [3];
    for(int i=0; i<list.size(); i++){
        row[0]= list.get(i).getAT();
        row[1]= list.get(i).getType();
        row[2]= list.get(i).getDiploma();

        model.addRow(row);
    }
}

public void executeQuery(String query, String message){
    Connection con = (new GetConnection()).getConnection();
    Statement st;

    try{
        st = con.createStatement();
        if((st.executeUpdate(query))==1){
            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();
            model.setRowCount(0);
            DisplayTable();
            JOptionPane.showMessageDialog(null, "Data "+message+"Suksesfully");
        }else{
            JOptionPane.showMessageDialog(null, "Data Not"+message);
        }
    }catch(Exception ex){
        ex.printStackTrace();
    }
}

@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();

    jTextField2 = new javax.swing.JTextField();

    jTextField3 = new javax.swing.JTextField();

    jButton1 = new javax.swing.JButton();

    jButton2 = new javax.swing.JButton();

    jButton3 = new javax.swing.JButton();

    jScrollPane1 = new javax.swing.JScrollPane();

    jTable1 = new javax.swing.JTable();

    jButton4 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    setLocation(new java.awt.Point(350, 200));

    jLabel1.setText("adm_AT");

    jLabel2.setText("adm_type");

    jLabel3.setText("adm_diploma");

    jButton1.setText("INSERT");
    jButton1.addActionListener(new java.awt.event.ActionListener() {

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

            jButton1ActionPerformed(evt);

        }

    });

    jButton2.setText("UPDATE");
    jButton2.addActionListener(new java.awt.event.ActionListener() {

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

            jButton2ActionPerformed(evt);

        }

    });

    jButton3.setText("DELETE");

```

```

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

        jButton3ActionPerformed(evt);

    }

});

```

```

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "adm_AT", "adm_type", "adm_diploma"

    }

) {

    boolean[] canEdit = new boolean [] {

        false, false, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

```

```

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {

    public void mouseClicked(java.awt.event.MouseEvent evt) {

        jTable1MouseClicked(evt);

    }

});

```

```

jScrollPane1.setViewportViewView(jTable1);

```

```

jButton4.setText("Menu");

jButton4.addActionListener(new java.awt.event.ActionListener() {

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

        jButton4ActionPerformed(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())

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

.addGroup(layout.createSequentialGroup())

.addGap(48, 48, 48)

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

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

.addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, 67, Short.MAX_VALUE))

.addComponent(jLabel3))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

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

.addComponent(jTextField3, javax.swing.GroupLayout.DEFAULT_SIZE, 109, Short.MAX_VALUE)

.addComponent(jTextField2)

.addComponent(jTextField1)))

.addGroup(layout.createSequentialGroup())

.addGap(41, 41, 41)

.addComponent(jButton1)

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

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

.addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

.addComponent(jButton3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))

.addGap(31, 31, 31)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

.addGap(34, 34, 34))

);

layout.setVerticalGroup(

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

.addGroup(layout.createSequentialGroup())

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

.addGroup(layout.createSequentialGroup())

.addGap(30, 30, 30)

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

.addComponent(jLabel1)

.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(jLabel2)

            .addComponent(jTextField2, 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(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGap(45, 45, 45)

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

            .addComponent(jButton1)

            .addComponent(jButton2)

            .addComponent(jButton3))

        .addGap(18, 18, 18)

        .addComponent(jButton4))

    .addGroup(layout.createSequentialGroup())

    .addGap(18, 18, 18)

    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 275, javax.swing.GroupLayout.PREFERRED_SIZE)))

    .addContainerGap(25, Short.MAX_VALUE))

);

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

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) // GEN-FIRST:event_jTable1MouseClicked
{
    DefaultTableModel model = (DefaultTableModel) jTable1.getModel();

    int selectedRowIndex = jTable1.getSelectedRow();

    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());
    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());
    jTextField3.setText(model.getValueAt(selectedRowIndex, 2).toString());
} // GEN-LAST:event_jTable1MouseClicked

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton1ActionPerformed
{
    String query = "INSERT INTO admin VALUES ('"+jTextField1.getText()+"','"+jTextField2.getText()+"','"+jTextField3.getText()+"')";

    executeQuery(query, "Inserted");
} // GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton2ActionPerformed

```

```

String query = "UPDATE admin SET adm_type='"+jTextField2.getText()+"',adm_diploma='"+jTextField3.getText()+"' WHERE adm_AT='"+jTextField1.getText()+"'";

executeQuery(query, "Updated");

} //GEN-LAST:event_jButton2ActionPerformed

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton3ActionPerformed

String query = "DELETE FROM admin WHERE adm_AT='"+jTextField1.getText()+"'";

executeQuery(query, "Delete");

} //GEN-LAST:event_jButton3ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton4ActionPerformed

this.setVisible(false);

new Menu().setVisible(true); //GEN-LAST:event_jButton4ActionPerformed

public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new Admin().setVisible(true);

        }

    });

}

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;

// End of variables declaration//GEN-END:variables
}

```

```
package bd;
```

```
import java.sql.*;
```

```
import java.util.ArrayList;
```

```
import javax.swing.*;
```

```
import javax.swing.table.*;
```

```
public class Branch extends javax.swing.JFrame {
```

```
    public Branch() {
```

```
        initComponents();
```

```
        this.setLocationRelativeTo(null);
```

```
        DisplayTable();
```

```
    }
```

```
    public ArrayList<SetterBranch> tableList(){
```

```
        ArrayList<SetterBranch> tableList = new ArrayList<>();
```

```
        Connection con = (new GetConnection()).getConnection();
```

```
        String query = "select * from branch ";
```

```
        Statement st;
```

```
        ResultSet rs;
```

```
        try{
```

```
            st = con.createStatement();
```

```
            rs = st.executeQuery(query);
```

```
            SetterBranch setter;
```

```
            while(rs.next()){
```

```
                setter = new SetterBranch(rs.getInt("br_code"), rs.getString("br_street"), rs.getInt("br_num"), rs.getString("br_city"));
```

```
                tableList.add(setter);
```

```
            }
```

```
        }catch(Exception ex){
```

```
            ex.printStackTrace();
```

```
        }
```

```
        return tableList;
```

```

}

private void DisplayTable(){
    ArrayList<SetterBranch> list = tableList();

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

    Object [] row = new Object [4];

    for(int i=0; i<list.size(); i++){

        row[0]= list.get(i).getCode();

        row[1]= list.get(i).getStreet();

        row[2]= list.get(i).getNum();

        row[3]= list.get(i).getCity();

        model.addRow(row);

    }
}

public void executeQuery(String query, String message){

    Connection con = (new GetConnection()).getConnection();

    Statement st;

    try{

        st = con.createStatement();

        if((st.executeUpdate(query))==1){

            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

            model.setRowCount(0);

            DisplayTable();

            JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");

        }else{

            JOptionPane.showMessageDialog(null, "Data Not"+message);

        }

    }catch(Exception ex){

        ex.printStackTrace();

    }

}

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

```

```

private void initComponents() {

    jTextField1 = new javax.swing.JTextField();

    jTextField2 = new javax.swing.JTextField();

    jTextField3 = new javax.swing.JTextField();

    jTextField4 = new javax.swing.JTextField();

    jButton1 = new javax.swing.JButton();

    jButton2 = new javax.swing.JButton();

    jButton3 = new javax.swing.JButton();

    jLabel1 = new javax.swing.JLabel();

    jLabel2 = new javax.swing.JLabel();

    jLabel3 = new javax.swing.JLabel();

    jLabel4 = new javax.swing.JLabel();

    jScrollPane2 = new javax.swing.JScrollPane();

    jTable1 = new javax.swing.JTable();

    jButton4 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    setLocation(new java.awt.Point(0, 0));

    jButton1.setText("INSERT");

    jButton1.addActionListener(new java.awt.event.ActionListener() {

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

            jButton1ActionPerformed(evt);

        }

    });

    jButton2.setText("UPDATE");

    jButton2.addActionListener(new java.awt.event.ActionListener() {

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

            jButton2ActionPerformed(evt);

        }

    });

    jButton3.setText("DELETE");

    jButton3.addActionListener(new java.awt.event.ActionListener() {

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

            jButton3ActionPerformed(evt);

        }

    });

```

```

});

jLabel1.setText("br_code");

jLabel2.setText("br_street");

jLabel3.setText("br_num");

jLabel4.setText("br_city");

JTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {

    },
    new String [] {
        "br_code", "br_street", "br_num", "br_city"
    }
) {
    boolean[] canEdit = new boolean [] {
        false, false, false, false
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return canEdit [columnIndex];
    }
});

JTable1.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        jTable1MouseClicked(evt);
    }
});

jScrollPane2.setViewportViewView(JTable1);

jButton4.setText("Menu");

jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton4ActionPerformed(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()

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

                .addGroup(layout.createSequentialGroup()

                    .add(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                        .add(layout.createSequentialGroup()

                            .addGap(43, 43, 43)

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

                                .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 58, javax.swing.GroupLayout.PREFERRED_SIZE)

                                .addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                .addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED_SIZE, 58, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                                .addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT_SIZE, 131, Short.MAX_VALUE)

                                .addComponent(jTextField2)

                                .addComponent(jTextField3)

                                .addComponent(jTextField4)))

                            .addGroup(layout.createSequentialGroup()

                                .add(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                                    .addContainerGap()

                                    .addComponent(jButton1, javax.swing.GroupLayout.PREFERRED_SIZE, 73, javax.swing.GroupLayout.PREFERRED_SIZE)

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

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

                                        .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                        .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

                                    .addComponent(jButton3)))

                                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 61, Short.MAX_VALUE)

                                .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, 658, javax.swing.GroupLayout.PREFERRED_SIZE)

                                .addContainerGap())

                                .addGap(58, 58, 58)

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

                                    .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

```



```

        .addComponent(jLabel1))

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

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

        .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(jLabel2))

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

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

        .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(jLabel3))

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

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

        .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addComponent(jLabel4))

    .addGap(33, 33, 33)

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

        .addComponent(jButton1)

        .addComponent(jButton2)

        .addComponent(jButton3))

    .addGap(18, 18, 18)

    .addComponent(jButton4)

    .addContainerGap(159, Short.MAX_VALUE))

    .addGroup(layout.createSequentialGroup())

    .addContainerGap()

    .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, 0, Short.MAX_VALUE)

    .addContainerGap())

);

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

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton1ActionPerformed
{
    String query = "INSERT INTO branch VALUES (NULL,'" + jTextField2.getText() + "','" + jTextField3.getText() + "','" + jTextField4.getText() + "')";
    executeQuery(query, "Inserted");
} // GEN-LAST:event_jButton1ActionPerformed

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton3ActionPerformed
{
    String query = "DELETE FROM branch WHERE br_code='" + jTextField1.getText() + "'";
    executeQuery(query, "Delete");
} // GEN-LAST:event_jButton3ActionPerformed

```

```

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {GEN-FIRST:event_jTable1MouseClicked

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

    int selectedRowIndex = jTable1.getSelectedRow();

    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());
    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());
    jTextField3.setText(model.getValueAt(selectedRowIndex, 2).toString());
    jTextField4.setText(model.getValueAt(selectedRowIndex, 3).toString());
}GEN-LAST:event_jTable1MouseClicked

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jButton2ActionPerformed

    String query = "UPDATE branch SET br_street='"+jTextField2.getText()+"',br_num='"+jTextField3.getText()+"',br_city='"+jTextField4.getText()+"'WHERE
br_code='"+jTextField1.getText()+"'";

    executeQuery(query, "Updated");
}GEN-LAST:event_jButton2ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jButton4ActionPerformed

    this.setVisible(false);

    new Menu().setVisible(true);
}GEN-LAST:event_jButton4ActionPerformed

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(Branch.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (InstantiationException ex) {

```

```

        java.util.logging.Logger.getLogger(Branch.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

        java.util.logging.Logger.getLogger(Branch.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

        java.util.logging.Logger.getLogger(Branch.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    }
}

//</editor-fold>
//</editor-fold>

/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {

    public void run() {

        new Branch().setVisible(true);

    }

});
}

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField4;

// End of variables declaration//GEN-END:variables
}

```

```

package bd;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.swing.table.DefaultTableModel;

public class BranchInfo extends javax.swing.JFrame {

    public BranchInfo() {

        initComponents();

        this.setLocationRelativeTo(null);

        tableList();

    }

    private void tableList(){

        Connection con = (new GetConnection()).getConnection();

        String query = "select br_code, br_street, br_num, br_city, wrk_lname, wrk_name, sum(res) as numofreservations, sum(tr_cost)*sum(res) as revenue\n" +
            "from (select br_code,br_street,      br_num, br_city, tr_id, tr_cost, count(*) as res\n" +
            "from branch inner join trip on br_code=tr_br_code\n" +
            "inner join reservation on tr_id=res_tr_id\n" +
            "group by tr_id) as t1\n" +
            "inner join manages on br_code=mng_br_code\n" +
            "inner join worker on mng_adm_AT=wrk_AT\n" +
            "group by br_code;";

        Statement st;

        ResultSet rs;

        try{

            st = con.createStatement();

            rs = st.executeQuery(query);

```

```

while(rs.next()){

    int data1 = rs.getInt("br_code");

    String data2 = rs.getString("br_street");

    int data3 = rs.getInt("br_num");

    String data4 = rs.getString("br_city");

    String data5 = rs.getString("wrk_lname");

    String data6 = rs.getString("wrk_name");

    int data7 = rs.getInt("numofreservations");

    int data8 = rs.getInt("revenue");


    Object[] row1 = { data1, data2, data3, data4, data5, data6, data7, data8};


    DefaultTableModel model1 = (DefaultTableModel) jTable1.getModel();


    model1.addRow(row1);

}

}catch(Exception ex){

    ex.printStackTrace();

}

}

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    jButton1 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jTable1.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {

        },

```



```

        .addComponent(jButton1)

        .addGap(18, 18, 18)

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 379, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGap(24, 24, 24))

    );

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

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton1ActionPerformed
{
    this.setVisible(false);

    new Menu().setVisible(true);
} // GEN-LAST:event_jButton1ActionPerformed


public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new BranchInfo().setVisible(true);

        }

    });
}

// Variables declaration - do not modify // GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
// End of variables declaration // GEN-END:variables
}

```

```
package bd;
```

```
import java.sql.Connection;
```

```

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.swing.table.DefaultTableModel;


public class BranchWorker extends javax.swing.JFrame {


    public BranchWorker() {

        initComponents();

        this.setLocationRelativeTo(null);

        tableList1();

        tableList2();

    }


    private void tableList1(){

        Connection con = (new GetConnection()).getConnection();

        String query = "select br_code, wrk_name, wrk_lname, wrk_salary\n" +
            "from branch right join worker on br_code=wrk_br_code\n" +
            "order by br_code;" ;

        Statement st;

        ResultSet rs;

        try{

            st = con.createStatement();

            rs = st.executeQuery(query);

            while(rs.next()){

                int data1 = rs.getInt("br_code");

                String data2 = rs.getString("wrk_lname");

                String data3 = rs.getString("wrk_name");

                float data4 = rs.getFloat("wrk_salary");

                Object[] row = { data1, data2, data3, data4};

```



```

DefaultTableModel model = (DefaultTableModel) jTable1.getModel();

        model.addRow(row);
    }

}

}catch(Exception ex){
    ex.printStackTrace();
}
}

private void tableList2(){

    Connection con = (new GetConnection()).getConnection();

    String query = "select br_code, sum(wrk_salary) as sumofsalary\n" +
        "from branch right join worker on br_code=wrk_br_code\n" +
        "group by br_code\n" +
        "order by br_code;";

    Statement st;
    ResultSet rs;

    try{

        st = con.createStatement();
        rs = st.executeQuery(query);

        while(rs.next()){

            String data1 = rs.getString("br_code");
            String data2 = rs.getString("sumofsalary");

            Object[] row = { data1, data2};

            DefaultTableModel model = (DefaultTableModel) jTable2.getModel();

            model.addRow(row);
        }

    }

}catch(Exception ex){

```

```

        ex.printStackTrace();
    }
}

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
private void initComponents() {

    JScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    JScrollPane2 = new javax.swing.JScrollPane();
    jTable2 = new javax.swing.JTable();
    jButton1 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jTable1.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {

        },
        new String [] {
            "br_code", "wrk_lname", "wrk_name", "wrk_salary"
        }
    ) {
        boolean[] canEdit = new boolean [] {
            false, false, true, false
        };
        public boolean isCellEditable(int rowIndex, int columnIndex) {
            return canEdit [columnIndex];
        }
    });
    jScrollPane1.setViewportView(jTable1);

    jTable2.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {

        },
        new String [] {

```

```

        "br_code", "sumofsalary"
    }
} {
    boolean[] canEdit = new boolean [] {

        false, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

jScrollPane2.setViewportViewView(jTable2);

jButton1.setText("Menu");
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()

            .addGap(18, 18, 18)

            .addComponent(jButton1)

            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

        .addGroup(layout.createSequentialGroup()

            .addGap(33, 33, 33)

            .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 375, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 86, Short.MAX_VALUE)

            .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, 314, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addGap(40, 40, 40))

);

layout.setVerticalGroup(

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

        .addGroup(layout.createSequentialGroup()

            .addGap(18, 18, 18)

```

```

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

        .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, 275, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGroup(layout.createSequentialGroup())

        .addComponent(jButton1)

        .addGap(32, 32, 32)

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 275, javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addContainerGap(79, Short.MAX_VALUE))

    );

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

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton1ActionPerformed
{
    this.setVisible(false);

    new Menu().setVisible(true);
} // GEN-LAST:event_jButton1ActionPerformed

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new BranchWorker().setVisible(true);

        }

    });
}

// Variables declaration - do not modify // GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JTable jTable1;
private javax.swing.JTable jTable2;

// End of variables declaration // GEN-END:variables
}

```

```
package bd;
```

```
public class ChooseTable extends javax.swing.JFrame {
```

```
    public ChooseTable() {  
        initComponents();  
        this.setLocationRelativeTo(null);  
    }
```

```
@SuppressWarnings("unchecked")
```

```
// <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-BEGIN: initComponents
```

```
private void initComponents() {
```

```
    buttonGroup1 = new javax.swing.ButtonGroup();  
    jRadioButtonBranch = new javax.swing.JRadioButton();  
    jRadioButtonWorker = new javax.swing.JRadioButton();  
    jRadioButtonAdmin = new javax.swing.JRadioButton();  
    jRadioButtonManages = new javax.swing.JRadioButton();  
    jRadioButtonPhones = new javax.swing.JRadioButton();  
    jRadioButtonDriver = new javax.swing.JRadioButton();  
    jRadioButtonGuide = new javax.swing.JRadioButton();  
    jRadioButtonLanguage = new javax.swing.JRadioButton();  
    jRadioButtonTrip = new javax.swing.JRadioButton();  
    jRadioButtonEvent = new javax.swing.JRadioButton();  
    jRadioButtonDestination = new javax.swing.JRadioButton();  
    jRadioButtonTravel_to = new javax.swing.JRadioButton();  
    jRadioButtonReservation = new javax.swing.JRadioButton();  
    jRadioButtonIT = new javax.swing.JRadioButton();  
    jRadioButtonOffers = new javax.swing.JRadioButton();  
    jRadioButtonReservation_offers = new javax.swing.JRadioButton();  
    jButton1 = new javax.swing.JButton();  
    jScrollPane2 = new javax.swing.JScrollPane();  
    jTextPane2 = new javax.swing.JTextPane();
```

```

jButton2 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

setAlwaysOnTop(true);

setAutoRequestFocus(false);

buttonGroup1.add(jRadioButtonBranch);
jRadioButtonBranch.setText("branch");
jRadioButtonBranch.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jRadioButtonBranchActionPerformed(evt);
    }
});

buttonGroup1.add(jRadioButtonWorker);
jRadioButtonWorker.setText("worker");

buttonGroup1.add(jRadioButtonAdmin);
jRadioButtonAdmin.setText("admin");

buttonGroup1.add(jRadioButtonManages);
jRadioButtonManages.setText("manages");

buttonGroup1.add(jRadioButtonPhones);
jRadioButtonPhones.setText("phones");

buttonGroup1.add(jRadioButtonDriver);
jRadioButtonDriver.setText("driver");
jRadioButtonDriver.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jRadioButtonDriverActionPerformed(evt);
    }
});

buttonGroup1.add(jRadioButtonGuide);
jRadioButtonGuide.setText("guide");

buttonGroup1.add(jRadioButtonLanguage);
jRadioButtonLanguage.setText("language");

```

```

buttonGroup1.add(jRadioButtonTrip);

jRadioButtonTrip.setText("trip");


buttonGroup1.add(jRadioButtonEvent);

jRadioButtonEvent.setText("event");

jRadioButtonEvent.addActionListener(new java.awt.event.ActionListener() {

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

        jRadioButtonEventActionPerformed(evt);

    }

});


buttonGroup1.add(jRadioButtonDestination);

jRadioButtonDestination.setText("destination");


buttonGroup1.add(jRadioButtonTravel_to);

jRadioButtonTravel_to.setText("travel_to");


buttonGroup1.add(jRadioButtonReservation);

jRadioButtonReservation.setText("reservation");


buttonGroup1.add(jRadioButtonIT);

jRadioButtonIT.setText("IT");


buttonGroup1.add(jRadioButtonOffers);

jRadioButtonOffers.setText("offers");


buttonGroup1.add(jRadioButtonReservation_offers);

jRadioButtonReservation_offers.setText("reservation_offers");

jRadioButtonReservation_offers.addActionListener(new java.awt.event.ActionListener() {

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

        jRadioButtonReservation_offersActionPerformed(evt);

    }

});


jButton1.setText("SELECT");

jButton1.addActionListener(new java.awt.event.ActionListener() {

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

        jButton1ActionPerformed(evt);

    }

});

```

$$\}));$$

```
jTextPane2.setFont(new java.awt.Font("sansserif", 3, 14)); // NOI18N
```

```
jScrollPane2.setViewportViewView(jTextPane2);
```

```
jButton2.addActionListener(new java.awt.event.ActionListener() {
```

```
jButton2ActionPerformed(evt);
```

$$\});$$

```
getContentPane().setLayout(layout);
```

```
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
.addComponent(jScrollPane2, javax.swing.GroupLayout.DEFAULT_SIZE, 474, Short.MAX_VALUE)
```

```
.addGroup(layout.createSequentialGroup())
```

```
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
.addGroup(layout.createSequentialGroup())
```

```
.addGap(60, 60, 60)
```

```
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
```

```
.addComponent(jRadioButtonManages, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addComponent(jRadioButtonAdmin, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addComponent(jRadioButtonWorker, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addComponent(jRadioButtonBranch, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
```

```
.addComponent(jRadioButtonEvent, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addComponent(jRadioButtonTrip, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
```

```
.addComponent(jRadioButtonTravel_to, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addComponent(jRadioButtonDestination, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE))))
```

```
.addGap(136, 136, 136)
```

```
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
.addComponent(jRadioButtonPhones, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)
```



```

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

            .addComponent(jRadioButtonGuide, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addComponent(jRadioButtonDriver, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE))

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

            .addComponent(jRadioButtonOffers, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addComponent(jRadioButtonIT, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addComponent(jRadioButtonReservation, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addComponent(jRadioButtonReservation_offers)

            .addComponent(jRadioButtonLanguage, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addGroup(layout.createSequentialGroup())

        .addGap(174, 174, 174)

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

            .addComponent(jButton1, javax.swing.GroupLayout.DEFAULT_SIZE, 93, Short.MAX_VALUE)

            .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))))

        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

);

layout.setVerticalGroup(

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

        .addGroup(layout.createSequentialGroup())

            .addGap(14, 14, 14)

            .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addGap(47, 47, 47)

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

                .addGroup(layout.createSequentialGroup())

                    .addComponent(jRadioButtonBranch)

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

                    .addComponent(jRadioButtonWorker)

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

                    .addComponent(jRadioButtonAdmin)

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

                    .addComponent(jRadioButtonManages))

                .addGroup(layout.createSequentialGroup())

                    .addComponent(jRadioButtonPhones)

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

                    .addComponent(jRadioButtonDriver)

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

                    .addComponent(jRadioButtonGuide)

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

                    .addComponent(jRadioButtonLanguage)))

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

```

```

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(layout.createSequentialGroup())

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

.addComponent(jRadioButtonTrip)

.addGroup(layout.createSequentialGroup())

.addGap(24, 24, 24)

.addComponent(jRadioButtonEvent)))

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

.addComponent(jRadioButtonDestination)

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

.addComponent(jRadioButtonTravel_to))

.addGroup(layout.createSequentialGroup())

.addComponent(jRadioButtonReservation)

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

.addComponent(jRadioButtonIT)

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

.addComponent(jRadioButtonOffers)

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

.addComponent(jRadioButtonReservation_offers)))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 45, Short.MAX_VALUE)

.addComponent(jButton1)

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

.addComponent(jButton2)

.addGap(28, 28, 28))

);

pack();

}

// </editor-fold> //GEN-END: initComponents

private void jRadioButtonDriverActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jRadioButtonDriverActionPerformed

    // TODO add your handling code here:

} //GEN-LAST:event_jRadioButtonDriverActionPerformed

private void jRadioButtonBranchActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jRadioButtonBranchActionPerformed

    // TODO add your handling code here:

} //GEN-LAST:event_jRadioButtonBranchActionPerformed

private void jRadioButtonEventActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jRadioButtonEventActionPerformed

    // TODO add your handling code here:

```

```
}//GEN-LAST:event_jRadioButtonEventActionPerformed
```

```
private void jRadioButtonReservation_offersActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jRadioButtonReservation_offersActionPerformed
```

```
    // TODO add your handling code here:
```

```
}//GEN-LAST:event_jRadioButtonReservation_offersActionPerformed
```

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton1ActionPerformed
```

```
    if(jRadioButtonBranch.isSelected()==true){
```

```
        this.setVisible(false);
```

```
        new Branch().setVisible(true);
```

```
    }
```

```
    if(jRadioButtonWorker.isSelected()==true){
```

```
        this.setVisible(false);
```

```
        new Worker().setVisible(true);
```

```
    }
```

```
    if(jRadioButtonAdmin.isSelected()==true){
```

```
        this.setVisible(false);
```

```
        new Admin().setVisible(true);
```

```
    }
```

```
    if(jRadioButtonManages.isSelected()==true){
```

```
        this.setVisible(false);
```

```
        new Manages().setVisible(true);
```

```
    }
```

```
    if(jRadioButtonPhones.isSelected()==true){
```

```
        this.setVisible(false);
```

```
        new Phones().setVisible(true);
```

```
    }
```

```
    if(jRadioButtonDriver.isSelected()==true){
```

```
        this.setVisible(false);
```

```
        new Driver().setVisible(true);
```

```
    }
```

```
    if(jRadioButtonGuide.isSelected()==true){
```

```
        this.setVisible(false);
```

```
        new Guide().setVisible(true);
```

```
    }
```

```
    if(jRadioButtonLanguage.isSelected()==true){
```

```
        this.setVisible(false);
```

```
        new Language().setVisible(true);
```

```
    }
```

```

if(jRadioButtonTrip.isSelected()==true){

    this.setVisible(false);

    new Trip().setVisible(true);

}

if(jRadioButtonEvent.isSelected()==true){

    this.setVisible(false);

    new Event().setVisible(true);

}

if(jRadioButtonDestination.isSelected()==true){

    this.setVisible(false);

    new Destination().setVisible(true);

}

if(jRadioButtonTravel_to.isSelected()==true){

    this.setVisible(false);

    new Travel_to().setVisible(true);

}

if(jRadioButtonReservation.isSelected()==true){

    this.setVisible(false);

    new Reservation().setVisible(true);

}

if(jRadioButtonIT.isSelected()==true){

    this.setVisible(false);

    new IT().setVisible(true);

}

if(jRadioButtonOffers.isSelected()==true){

    this.setVisible(false);

    new Offers().setVisible(true);

}

if(jRadioButtonReservation_offers.isSelected()==true){

    this.setVisible(false);

    new Reservation_offers().setVisible(true);

}

}

//GEN-LAST:event_jButton1ActionPerformed

```

```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jButton2ActionPerformed

    this.setVisible(false);

    new Menu().setVisible(true);

```

```
//GEN-LAST:event_jButton2ActionPerformed
```

```
public static void main(String args[]) {
```

```
    java.awt.EventQueue.invokeLater(new Runnable() {  
        public void run() {  
            new ChooseTable().setVisible(true);  
        }  
    });  
}
```

```
// Variables declaration - do not modify//GEN-BEGIN:variables
```

```
private javax.swing.ButtonGroup buttonGroup1;  
private javax.swing.JButton jButton1;  
private javax.swing.JButton jButton2;  
private javax.swing.JRadioButton jRadioButtonAdmin;  
private javax.swing.JRadioButton jRadioButtonBranch;  
private javax.swing.JRadioButton jRadioButtonDestination;  
private javax.swing.JRadioButton jRadioButtonDriver;  
private javax.swing.JRadioButton jRadioButtonEvent;  
private javax.swing.JRadioButton jRadioButtonGuide;  
private javax.swing.JRadioButton jRadioButtonIT;  
private javax.swing.JRadioButton jRadioButtonLanguage;  
private javax.swing.JRadioButton jRadioButtonManages;  
private javax.swing.JRadioButton jRadioButtonOffers;  
private javax.swing.JRadioButton jRadioButtonPhones;  
private javax.swing.JRadioButton jRadioButtonReservation;  
private javax.swing.JRadioButton jRadioButtonReservation_offers;  
private javax.swing.JRadioButton jRadioButtonTravel_to;  
private javax.swing.JRadioButton jRadioButtonTrip;  
private javax.swing.JRadioButton jRadioButtonWorker;  
private javax.swing.JScrollPane jScrollPane2;  
private javax.swing.JTextPane jTextPane2;  
  
// End of variables declaration//GEN-END:variables  
}
```

```
package bd;
```

```
import java.sql.Connection;
```

```
import java.sql.DriverManager;
```

```
import java.sql.ResultSet;
```

```
import java.sql.SQLException;
```

```
import java.sql.Statement;
```

```
import javax.swing.table.DefaultTableModel;
```

```
public class CustomerReservation extends javax.swing.JFrame {
```

```
    public CustomerReservation() {
```

```
        initComponents();
```

```
        this.setLocationRelativeTo(null);
```

```
    }
```

```
    private void tableList(){
```

```
        Connection con = (new GetConnection()).getConnection();
```

```
        String query = "call customerReservation('"+textField1.getText()+"')";
```

```
        Statement st;
```

```
        ResultSet rs;
```

```
        try{
```

```
            st = con.createStatement();
```

```
            rs = st.executeQuery(query);
```

```
            while(rs.next()){
```

```
                String data1 = rs.getString("res_off_ln");
```

```
                String data2 = rs.getString("res_off_fn");
```

```
                int data3 = rs.getInt("res_off_id");
```

```

int data4 = rs.getInt("count(*)");

Object[] row1 = { data1, data2, data3, data4};

DefaultTableModel model1 = (DefaultTableModel) jTable1.getModel();

model1.addRow(row1);
}

}catch(Exception ex){
    ex.printStackTrace();
}
}

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN: initComponents
private void initComponents() {

    jLabel1 = new javax.swing.JLabel();
    jTextField1 = new javax.swing.JTextField();
    jButton1 = new javax.swing.JButton();
    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    jButton2 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel1.setText("Last Name");

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

    jButton1.setText("SELECT");

```



```

        .addComponent(jLabel1)

        .addGap(18, 18, 18)

        .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, 131, javax.swing.GroupLayout.PREFERRED_SIZE))

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

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 97, javax.swing.GroupLayout.PREFERRED_SIZE)

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

        .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

        .addComponent(jButton1))

    .addGap(36, 36, 36)))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 44, Short.MAX_VALUE)

    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 618, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addGap(16, 16, 16))

);

layout.setVerticalGroup(

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

    .addGroup(layout.createSequentialGroup())

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

        .addGroup(layout.createSequentialGroup())

        .addGap(17, 17, 17)

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 300, javax.swing.GroupLayout.PREFERRED_SIZE))

    .addGroup(layout.createSequentialGroup())

        .addGap(102, 102, 102)

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

            .addComponent(jLabel1)

            .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGap(29, 29, 29)

        .addComponent(jButton1)

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

        .addComponent(jButton2)))

    .addContainerGap(40, Short.MAX_VALUE))

);

pack();

} // </editor-fold> // GEN-END: initComponents

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton1ActionPerformed

    tableList();

} // GEN-LAST:event_jButton1ActionPerformed

```

```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jButton2ActionPerformed

    this.setVisible(false);

    new Menu().setVisible(true);

}GEN-LAST:event_jButton2ActionPerformed


private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jTextField1ActionPerformed

    // TODO add your handling code here:

}GEN-LAST:event_jTextField1ActionPerformed


public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new CustomerReservation().setVisible(true);

        }

    });

}


// Variables declaration - do not modifyGEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JLabel jLabel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;

// End of variables declarationGEN-END:variables
}

```

```

package bd;

import java.sql.*;
import java.util.ArrayList;
import javax.swing.*;
import javax.swing.table.*;

```

```

public class Destination extends javax.swing.JFrame {

    public Destination() {

        initComponents();

        this.setLocationRelativeTo(null);

        DisplayTable();

    }


    public ArrayList<SetterDestination> tableList(){

        ArrayList<SetterDestination> tableList = new ArrayList<>();

        Connection con = (new GetConnection()).getConnection();

        String query = "select * from destination ";

        Statement st;

        ResultSet rs;

        try{

            st = con.createStatement();

            rs = st.executeQuery(query);

            SetterDestination setter;

            while(rs.next()){

                setter = new SetterDestination(rs.getInt("dst_id"),rs.getString("dst_name"),rs.getString("dst_descr"),rs.getString("dst_rtype"), rs.getString("dst_language"),
rs.getInt("dst_location"));

                tableList.add(setter);

            }

        }catch(Exception ex){

            ex.printStackTrace();

        }

        return tableList;

    }


    private void DisplayTable(){

        ArrayList<SetterDestination> list = tableList();

        DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

        Object [] row = new Object [6];

        for(int i=0; i<list.size(); i++){

```

```

row[0]= list.get(i).getID();

row[1]= list.get(i).getName();

row[2]= list.get(i).getDescr();

row[3]= list.get(i).getRtype();

row[4]= list.get(i).getLanguage();

if(list.get(i).getLocation() == 0){

    row[5]= "NULL";

}else{

row[5]= list.get(i).getLocation();

}

model.addRow(row);

}

}

```

```

public void executeQuery(String query, String message){

    Connection con = (new GetConnection()).getConnection();

    Statement st;

    try{

        st = con.createStatement();

        if((st.executeUpdate(query))==1){

            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

            model.setRowCount(0);

            DisplayTable();

            JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");

        }else{

            JOptionPane.showMessageDialog(null, "Data Not"+message);

        }

    }catch(Exception ex){

        ex.printStackTrace();

    }

}

```

```
@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();

jLabel4 = new javax.swing.JLabel();

jLabel5 = new javax.swing.JLabel();

jLabel6 = new javax.swing.JLabel();

jTextField1 = new javax.swing.JTextField();

jTextField2 = new javax.swing.JTextField();

jTextField3 = new javax.swing.JTextField();

jTextField4 = new javax.swing.JTextField();

jTextField5 = new javax.swing.JTextField();

jTextField6 = new javax.swing.JTextField();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

jButton3 = new javax.swing.JButton();

jScrollPane1 = new javax.swing.JScrollPane();

jTable1 = new javax.swing.JTable();

jButton4 = new javax.swing.JButton();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);


jLabel1.setText("dst_tr_id");


jLabel2.setText("dst_name");


jLabel3.setText("dst_descr");


jLabel4.setText("dst_rtype");


jLabel5.setText("dst_language");


jLabel6.setText("dst_location");


jButton1.setText("INSERT");

jButton1.addActionListener(new java.awt.event.ActionListener() {

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

        jButton1ActionPerformed(evt);

    }

});

```

```

jButton2.setText("UPDATE");

jButton2.addActionListener(new java.awt.event.ActionListener() {

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

        jButton2ActionPerformed(evt);

    }

});

jButton3.setText("DELETE");

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

        jButton3ActionPerformed(evt);

    }

});

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "dst_tr_id", "dst_name", "dst_descr", "dst_rtype", "dst_language", "dst_location"

    }

) {

    boolean[] canEdit = new boolean [] {

        false, false, false, false, true, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {

    public void mouseClicked(java.awt.event.MouseEvent evt) {

        jTable1MouseClicked(evt);

    }

});

jScrollPane1.setViewportViewView(jTable1);

jButton4.setText("Menu");

```

```

jButton4.addActionListener(new java.awt.event.ActionListener() {

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

        jButton4ActionPerformed(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()

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

                .addGroup(layout.createSequentialGroup()

                    .addGroup(layout.createSequentialGroup()

                        .addGroup(layout.createSequentialGroup()

                            .addGroup(layout.createSequentialGroup()

                                .addGap(31, 31, 31)

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

                                    .addGroup(layout.createSequentialGroup()

                                        .addGroup(layout.createSequentialGroup()

                                            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

                                                .addComponent(jLabel2, javax.swing.GroupLayout.Alignment.LEADING, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                                .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

                                    .addGap(53, 53, 53))

                                .addGroup(layout.createSequentialGroup()

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

                                        .addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                        .addComponent(jLabel4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                        .addComponent(jLabel5, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                        .addComponent(jLabel6, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

                                    .addGap(33, 33, 33)))

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

                                    .addComponent(jTextField1)

                                    .addComponent(jTextField2)

                                    .addComponent(jTextField3)

                                    .addComponent(jTextField4)

                                    .addComponent(jTextField5)

                                    .addComponent(jTextField6)))

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

                                    .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                    .addComponent(jButton1)

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

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

```

```

        .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

        .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

    .addComponent(jButton3)))

.addGap(40, 40, 40)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 867, javax.swing.GroupLayout.PREFERRED_SIZE)

.addGap(22, 22, 22))

);

layout.setVerticalGroup(

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

    .addGroup(layout.createSequentialGroup())

    .addGap(22, 22, 22)

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

    .addGroup(layout.createSequentialGroup())

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

    .addComponent(jLabel1)

    .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

    .addComponent(jLabel2)

    .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

    .addComponent(jLabel3)

    .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

    .addComponent(jLabel4)

    .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

    .addComponent(jLabel5)

    .addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

    .addComponent(jLabel6)

    .addComponent(jTextField6, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

    .addGap(44, 44, 44)

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

```



```

        .addComponent(jButton1)

        .addComponent(jButton2)

        .addComponent(jButton3))

    .addGap(18, 18, 18)

    .addComponent(jButton4))

    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 464, javax.swing.GroupLayout.PREFERRED_SIZE))

    .addContainerGap(31, Short.MAX_VALUE))

);

pack();

} // </editor-fold> // GEN-END: initComponents

```

```

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) // GEN-FIRST:event_jTable1MouseClicked

```

```

    DefaultTableModel model = (DefaultTableModel) jTable1.getModel();

```

```

    int selectedRowIndex = jTable1.getSelectedRow();

```

```

    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());

```

```

    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());

```

```

    jTextField3.setText(model.getValueAt(selectedRowIndex, 2).toString());

```

```

    jTextField4.setText(model.getValueAt(selectedRowIndex, 3).toString());

```

```

    jTextField5.setText(model.getValueAt(selectedRowIndex, 4).toString());

```

```

    jTextField6.setText(model.getValueAt(selectedRowIndex, 5).toString());

```

```

} // GEN-LAST:event_jTable1MouseClicked

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton1ActionPerformed

```

```

    String query = "INSERT INTO destination VALUES ( NULL, '"+jTextField2.getText()+"', '"+jTextField3.getText()+"', '"+jTextField4.getText()+"',
    '"+jTextField5.getText()+"', '"+jTextField6.getText()+"');";

```

```

    executeQuery(query, "Inserted");

```

```

} // GEN-LAST:event_jButton1ActionPerformed

```

```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton2ActionPerformed

```

```

    String query = "UPDATE destination SET dst_name= '"+jTextField2.getText()+"', dst_descr= '"+jTextField3.getText()+"', dst_rtype= '"+jTextField4.getText()+"', dst_language=
    '"+jTextField5.getText()+"', dst_location= '"+jTextField6.getText()+"' WHERE dst_id= '"+jTextField1.getText()+"' ";

```

```

    executeQuery(query, "Updated");

```

```

} // GEN-LAST:event_jButton2ActionPerformed

```

```

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton3ActionPerformed

```

```

    String query = "DELETE FROM destination WHERE dst_id= '"+jTextField1.getText()+"' ";

```

```

    executeQuery(query, "Delete");

```

```

} //GEN-LAST:event_jButton3ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton4ActionPerformed
{
    this.setVisible(false);

    new Menu().setVisible(true);
} //GEN-LAST:event_jButton4ActionPerformed


public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new Destination().setVisible(true);

        }

    });
}


// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField4;
private javax.swing.JTextField jTextField5;
private javax.swing.JTextField jTextField6;
// End of variables declaration//GEN-END:variables
}

```

```
package bd;
```

```
import java.sql.*;
```

```
import java.util.ArrayList;
```

```
import javax.swing.*;
```

```
import javax.swing.table.*;
```

```
public class Driver extends javax.swing.JFrame {
```

```
    public Driver() {
```

```
        initComponents();
```

```
        this.setLocationRelativeTo(null);
```

```
        DisplayTable();
```

```
    }
```

```
    public ArrayList<SetterDriver> tableList(){
```

```
        ArrayList<SetterDriver> tableList = new ArrayList<>();
```

```
        Connection con = (new GetConnection()).getConnection();
```

```
        String query = "select * from driver ";
```

```
        Statement st;
```

```
        ResultSet rs;
```

```
        try{
```

```
            st = con.createStatement();
```

```
            rs = st.executeQuery(query);
```

```
            SetterDriver setter;
```

```
            while(rs.next()){
```

```
                setter = new SetterDriver(rs.getString("drv_AT"),rs.getString("drv_license"),rs.getString("drv_route"),rs.getInt("drv_experience"));
```

```
                tableList.add(setter);
```

```
            }
```

```

}catch(Exception ex){

    ex.printStackTrace();

}

return tableList;
}

private void DisplayTable(){

ArrayList<SetterDriver> list = tableList();

DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

Object [] row = new Object [4];

for(int i=0; i<list.size(); i++){

    row[0]= list.get(i).getAT();

    row[1]= list.get(i).getLicense();

    row[2]= list.get(i).getRoute();

    row[3]= list.get(i).getExperience();


    model.addRow(row);

}

}

public void executeQuery(String query, String message){

    Connection con = (new GetConnection()).getConnection();

    Statement st;

    try{

        st = con.createStatement();

        if((st.executeUpdate(query))!=1){

            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

            model.setRowCount(0);

            DisplayTable();

            JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");

        }else{

            JOptionPane.showMessageDialog(null, "Data Not"+message);

        }

    }catch(Exception ex){

        ex.printStackTrace();

    }
}

```

```
}
```

```
@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();
```

```
    jLabel4 = new javax.swing.JLabel();
```

```
    jTextField1 = new javax.swing.JTextField();
```

```
    jTextField2 = new javax.swing.JTextField();
```

```
    jTextField3 = new javax.swing.JTextField();
```

```
    jTextField4 = new javax.swing.JTextField();
```

```
    jButton1 = new javax.swing.JButton();
```

```
    jButton2 = new javax.swing.JButton();
```

```
    jButton3 = new javax.swing.JButton();
```

```
    jScrollPane1 = new javax.swing.JScrollPane();
```

```
    jTable1 = new javax.swing.JTable();
```

```
    jButton4 = new javax.swing.JButton();
```

```
    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
```

```
    jLabel1.setText("drv_AT");
```

```
    jLabel2.setText("drv_license");
```

```
    jLabel3.setText("drv_route");
```

```
    jLabel4.setText("drv_experience");
```

```
    jButton1.setText("INSERT");
```

```
    jButton1.addActionListener(new java.awt.event.ActionListener() {
```

```
        public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
            jButton1ActionPerformed(evt);
```

```
        }
```

```
    });
```

```
    jButton2.setText("UPDATE");
```

```

jButton2.addActionListener(new java.awt.event.ActionListener() {

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

        jButton2ActionPerformed(evt);

    }

});

```

```

jButton3.setText("DELETE");

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

        jButton3ActionPerformed(evt);

    }

});

```

```

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "drv_AT", "drv_license", "drv_route", "drv_experience"

    }

) {

    boolean[] canEdit = new boolean [] {

        false, false, true, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

```

```

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {

    public void mouseClicked(java.awt.event.MouseEvent evt) {

        jTable1MouseClicked(evt);

    }

});

```

```

jScrollPane1.setViewportView(jTable1);

```

```

jButton4.setText("Menu");

jButton4.addActionListener(new java.awt.event.ActionListener() {

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

```

```

jButton4ActionPerformed(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()

            .addGap(55, 55, 55)

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

                .addGroup(layout.createSequentialGroup()

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

                        .addGroup(layout.createSequentialGroup()

                            .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                            .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                            .addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED_SIZE, 58, javax.swing.GroupLayout.PREFERRED_SIZE))

                        .addComponent(jLabel4))

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)

                        .addComponent(jTextField3)

                        .addComponent(jTextField2, javax.swing.GroupLayout.Alignment.LEADING)

                        .addComponent(jTextField1, javax.swing.GroupLayout.Alignment.LEADING)

                        .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, 134, javax.swing.GroupLayout.PREFERRED_SIZE)))

                    .addGroup(layout.createSequentialGroup()

                        .addComponent(jButton1)

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

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

                            .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                            .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

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

                            .addComponent(jButton3)))

                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 53, Short.MAX_VALUE)

                        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 525, javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addGap(17, 17, 17))

                .addGap(17, 17, 17))

            .addContainerGap(17, true))

    );

layout.setVerticalGroup(

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

        .addGroup(layout.createSequentialGroup()

            .addContainerGap()

            .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 525, javax.swing.GroupLayout.PREFERRED_SIZE)

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

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

                .addGroup(layout.createSequentialGroup()

                    .addComponent(jButton3)

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

                    .addComponent(jButton2)

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

                    .addComponent(jButton4)

                    .addContainerGap(17, true))

                .addGroup(layout.createSequentialGroup()

                    .addComponent(jButton1)

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

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

                        .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, 134, javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addComponent(jTextField1, javax.swing.GroupLayout.Alignment.LEADING)

                        .addComponent(jTextField2, javax.swing.GroupLayout.Alignment.LEADING)

                        .addComponent(jTextField3)

                        .addContainerGap())

                    .addComponent(jLabel4)

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

                        .addGroup(layout.createSequentialGroup()

                            .addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED_SIZE, 58, javax.swing.GroupLayout.PREFERRED_SIZE)

                            .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                            .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

                        .addContainerGap())

                    .addGap(55, 55, 55))

            .addContainerGap(17, true))

    );

```

```

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

.addGroup(layout.createSequentialGroup())

.addGap(15, 15, 15)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 350, javax.swing.GroupLayout.PREFERRED_SIZE))

.addGroup(layout.createSequentialGroup())

.addGap(59, 59, 59)

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

.addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(jLabel1))

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

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

.addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(jLabel2))

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

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

.addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(jLabel3))

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

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

.addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(jLabel4))

.addGap(59, 59, 59)

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

.addComponent(jButton1)

.addComponent(jButton2)

.addComponent(jButton3))

.addGap(18, 18, 18)

.addComponent(jButton4)))

.addContainerGap(15, Short.MAX_VALUE))

);

pack();

} // </editor-fold> // GEN-END: initComponents

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) // GEN-FIRST:event_jTable1MouseClicked

DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

int selectedRowIndex = jTable1.getSelectedRow();

```



```

        jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());

        jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());

        jTextField3.setText(model.getValueAt(selectedRowIndex, 2).toString());

        jTextField4.setText(model.getValueAt(selectedRowIndex, 3).toString());

    } //GEN-LAST:event_jTable1MouseClicked

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton1ActionPerformed

    String query = "INSERT INTO driver VALUES ( '"+jTextField1.getText()+"', '"+jTextField2.getText()+"', '"+jTextField3.getText()+"', '"+jTextField4.getText()+" ');";

    executeQuery(query, "Inserted");

} //GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton2ActionPerformed

    String query = "UPDATE driver SET drv_license= '"+jTextField2.getText()+"', drv_route= '"+jTextField3.getText()+"', drv_experience='"+jTextField4.getText()+"' WHERE drv_AT= '"+jTextField1.getText()+"' ";

    executeQuery(query, "Updated");

} //GEN-LAST:event_jButton2ActionPerformed

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton3ActionPerformed

    String query = "DELETE FROM driver WHERE drv_AT= '"+jTextField1.getText()+"' ";

    executeQuery(query, "Delete");

} //GEN-LAST:event_jButton3ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton4ActionPerformed

    this.setVisible(false);

    new Menu().setVisible(true); } //GEN-LAST:event_jButton4ActionPerformed

public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new Driver().setVisible(true);

        }

    });

}

// Variables declaration - do not modify //GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;

```

```

private javax.swing.JButton jButton4;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTable jTable1;

private javax.swing.JTextField jTextField1;

private javax.swing.JTextField jTextField2;

private javax.swing.JTextField jTextField3;

private javax.swing.JTextField jTextField4;

// End of variables declaration//GEN-END:variables
}

```

```

package bd;

```

```

import java.sql.*;

import java.util.ArrayList;

import javax.swing.*;

import javax.swing.table.*;

```

```

public class Event extends javax.swing.JFrame {

```

```

    public Event() {

        initComponents();

        this.setLocationRelativeTo(null);

        DisplayTable();

    }

```

```

    public ArrayList<SetterEvent> tableList(){

```

```

        ArrayList<SetterEvent> tableList = new ArrayList<>();

```

```

Connection con = (new GetConnection()).getConnection();

String query = "select * from event ";

Statement st;

ResultSet rs;


try{

    st = con.createStatement();

    rs = st.executeQuery(query);

    SetterEvent setter;

    while(rs.next()){

        setter = new SetterEvent(rs.getInt("ev_tr_id"),rs.getString("ev_start"),rs.getString("ev_end"),rs.getString("ev_descr"));

        tableList.add(setter);

    }

}catch(Exception ex){

    ex.printStackTrace();

}

return tableList;

}

```

```

private void DisplayTable(){

ArrayList<SetterEvent> list = tableList();

DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

Object [] row = new Object [4];

for(int i=0; i<list.size(); i++){

    row[0]= list.get(i).getID();

    row[1]= list.get(i).getStart();

    row[2]= list.get(i).getEnd();

    row[3]= list.get(i).getDescr();


    model.addRow(row);

}

}

```

```

public void executeQuery(String query, String message){

    Connection con = (new GetConnection()).getConnection();

    Statement st;


    try{

```

```

st = con.createStatement();

if((st.executeUpdate(query))==1){

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

    model.setRowCount(0);

    DisplayTable();

    JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");

}else{

    JOptionPane.showMessageDialog(null, "Data Not"+message);

}

}catch(Exception ex){

    ex.printStackTrace();

}

}

@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();

    jLabel4 = new javax.swing.JLabel();

    jTextField1 = new javax.swing.JTextField();

    jTextField2 = new javax.swing.JTextField();

    jTextField3 = new javax.swing.JTextField();

    jTextField4 = new javax.swing.JTextField();

    jButton1 = new javax.swing.JButton();

    jButton2 = new javax.swing.JButton();

    jButton3 = new javax.swing.JButton();

    jScrollPane1 = new javax.swing.JScrollPane();

    jTable1 = new javax.swing.JTable();

    jButton4 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel1.setText("ev_tr_id");

    jLabel2.setText("ev_start");

```

```

jLabel3.setText("ev_end");

jLabel4.setText("ev_descr");

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

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

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

jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {

    },
    new String [] {
        "ev_tr_id", "ev_start", "ev_end", "ev_descr"
    }
) {
    boolean[] canEdit = new boolean [] {
        false, false, true, false
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

```



```

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

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

            .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

            .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

        .addGap(12, 12, 12)

        .addComponent(jButton3)))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 45, Short.MAX_VALUE)

    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 532, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addGap(25, 25, 25))

);

layout.setVerticalGroup(

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

    .addGroup(layout.createSequentialGroup())

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

        .addGroup(layout.createSequentialGroup())

            .addGap(39, 39, 39)

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

                .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                .addComponent(jLabel1))

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

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

                .addComponent(jLabel2)

                .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                .addComponent(jLabel3)

                .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                .addComponent(jLabel4)

                .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

            .addGap(33, 33, 33)

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

                .addComponent(jButton1)

                .addComponent(jButton2)

                .addComponent(jButton3))

            .addGap(18, 18, 18)

            .addComponent(jButton4))

    .addGroup(layout.createSequentialGroup())

```

```

        .addGap(16, 16, 16)

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 326, javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addContainerGap(18, Short.MAX_VALUE))

    };

    pack();
}

// </editor-fold> //GEN-END: initComponents

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) { //GEN-FIRST:event_jTable1MouseClicked

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

    int selectedRowIndex = jTable1.getSelectedRow();

    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());
    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());
    jTextField3.setText(model.getValueAt(selectedRowIndex, 2).toString());
    jTextField4.setText(model.getValueAt(selectedRowIndex, 3).toString());
} //GEN-LAST:event_jTable1MouseClicked

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton1ActionPerformed

    String query = "INSERT INTO event VALUES ( "+jTextField1.getText()+" , "+jTextField2.getText()+" , "+jTextField3.getText()+" , "+jTextField4.getText()+" )";

    executeQuery(query, "Inserted");
} //GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton2ActionPerformed

    String query = "UPDATE event SET ev_start= '"+jTextField2.getText()+"' , ev_end= '"+jTextField3.getText()+"' , ev_descr= '"+jTextField4.getText()+"' WHERE ev_tr_id= "+jTextField1.getText()+" ";

    executeQuery(query, "Updated");
} //GEN-LAST:event_jButton2ActionPerformed

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton3ActionPerformed

    String query = "DELETE FROM event WHERE ev_tr_id= "+jTextField1.getText()+" ";

    executeQuery(query, "Delete");
} //GEN-LAST:event_jButton3ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton4ActionPerformed

    this.setVisible(false);

    new Menu().setVisible(true);
} //GEN-LAST:event_jButton4ActionPerformed

```



```

public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new Event().setVisible(true);

        }

    });

}

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField4;

// End of variables declaration//GEN-END:variables
}

```

```

package bd;

```

```

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

```

```

public class GetConnection {

    public Connection getConnection(){

        String DB_URL = "jdbc:mysql://localhost:3306/travelagency";

        String USERNAME = "root";

        String PASSWORD = "root";

        Connection connection;

        try{

            connection = DriverManager.getConnection(DB_URL, USERNAME, PASSWORD);

            return connection;

        }catch(SQLException ex){

            ex.printStackTrace();

            return null;

        }

    }

}

```

```

package bd;

```

```

import java.sql.*;

import java.util.ArrayList;

import javax.swing.*;

import javax.swing.table.*;

```

```

public class Guide extends javax.swing.JFrame {

```

```

    public Guide() {

        initComponents();

        this.setLocationRelativeTo(null);

        DisplayTable();

    }

```

```

public ArrayList<SetterGuide> tableList(){

    ArrayList<SetterGuide> tableList = new ArrayList<>();

    Connection con = (new GetConnection()).getConnection();

    String query = "select * from guide ";

    Statement st;

    ResultSet rs;

    try{

        st = con.createStatement();

        rs = st.executeQuery(query);

        SetterGuide setter;

        while(rs.next()){

            setter = new SetterGuide(rs.getString("gui_AT"),rs.getString("gui_cv"));

            tableList.add(setter);

        }

    }catch(Exception ex){

        ex.printStackTrace();

    }

    return tableList;

}

```

```

private void DisplayTable(){

    ArrayList<SetterGuide> list = tableList();

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

    Object [] row = new Object [2];

    for(int i=0; i<list.size(); i++){

        row[0]= list.get(i).getAT();

        row[1]= list.get(i).getCV();

        model.addRow(row);

    }

}

```

```

public void executeQuery(String query, String message){

    Connection con = (new GetConnection()).getConnection();

    Statement st;

```

```

try{

    st = con.createStatement();

    if((st.executeUpdate(query))==1){

        DefaultTableModel model = (DefaultTableModel)Table1.getModel();

        model.setRowCount(0);

        DisplayTable();

        JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");

    }else{

        JOptionPane.showMessageDialog(null, "Data Not"+message);

    }

}catch(Exception ex){

    ex.printStackTrace();

}

}

@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();
    jTextField1 = new javax.swing.JTextField();
    jTextField2 = new javax.swing.JTextField();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();
    jButton3 = new javax.swing.JButton();
    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    jButton4 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel1.setText("gui_AT");

    jLabel2.setText("gui_cv");

    jButton1.setText("INSERT");

```

```

jButton1.addActionListener(new java.awt.event.ActionListener() {

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

        jButton1ActionPerformed(evt);

    }

});

```

```

jButton2.setText("UPDATE");

jButton2.addActionListener(new java.awt.event.ActionListener() {

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

        jButton2ActionPerformed(evt);

    }

});

```

```

jButton3.setText("DELETE");

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

        jButton3ActionPerformed(evt);

    }

});

```

```

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "gui_AT", "gui_cv"

    }

) {

    boolean[] canEdit = new boolean [] {

        true, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {

    public void mouseClicked(java.awt.event.MouseEvent evt) {

        jTable1MouseClicked(evt);

    }

});

```



```

layout.setVerticalGroup(

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

        .addGroup(layout.createSequentialGroup())

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

                .addGroup(layout.createSequentialGroup())

                    .addGap(29, 29, 29)

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

                        .addComponent(jLabel1)

                        .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                        .addComponent(jLabel2)

                        .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addGap(41, 41, 41)

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

                        .addComponent(jButton1)

                        .addComponent(jButton2)

                        .addComponent(jButton3))

                    .addGap(18, 18, 18)

                    .addComponent(jButton4))

            .addGroup(layout.createSequentialGroup())

                .addGap(21, 21, 21)

                .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 292, javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addContainerGap(28, Short.MAX_VALUE))

);

pack();
}

// </editor-fold> //GEN-END: initComponents

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) { //GEN-FIRST:event_jTable1MouseClicked

    DefaultTableModel model = (DefaultTableModel) jTable1.getModel();

    int selectedRowIndex = jTable1.getSelectedRow();

    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());

    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());

}

//GEN-LAST:event_jTable1MouseClicked

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton1ActionPerformed

```

```

String query = "INSERT INTO guide VALUES ( '"+jTextField1.getText()+"', '"+jTextField2.getText()+"' );";

executeQuery(query, "Inserted");

} //GEN-LAST:event_jButton1ActionPerformed


private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton2ActionPerformed

String query = "UPDATE guide SET gui_cv= '"+jTextField2.getText()+"' WHERE gui_AT= '"+jTextField1.getText()+"' ";

executeQuery(query, "Updated");

} //GEN-LAST:event_jButton2ActionPerformed


private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton3ActionPerformed

String query = "DELETE FROM guide WHERE gui_AT= '"+jTextField1.getText()+"' ";

executeQuery(query, "Delete");

} //GEN-LAST:event_jButton3ActionPerformed


private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton4ActionPerformed

this.setVisible(false);

new Menu().setVisible(true); } //GEN-LAST:event_jButton4ActionPerformed


public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new Guide().setVisible(true);

        }

    });

}


// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;

```



```
// End of variables declaration//GEN-END:variables  
}
```

```
package bd;
```

```
import java.sql.*;  
import java.util.ArrayList;  
import javax.swing.*.*;  
import javax.swing.table.*;
```

```
public class IT extends javax.swing.JFrame {
```

```
    public IT() {  
        initComponents();  
        this.setLocationRelativeTo(null);  
        DisplayTable();  
    }
```

```
    public ArrayList<SetterIT> tableList(){
```

```
        ArrayList<SetterIT> tableList = new ArrayList<>();  
        Connection con = (new GetConnection()).getConnection();  
        String query = "select * from IT ";  
        Statement st;  
        ResultSet rs;
```

```
        try{  
            st = con.createStatement();  
            rs = st.executeQuery(query);  
            SetterIT setter;  
            while(rs.next()){
```

```

        setter = new SetterIT(rs.getString("IT_AT"),rs.getString("password"), rs.getString("start_date"), rs.getString("end_date"));

        tableList.add(setter);

    }

}catch(Exception ex){

    ex.printStackTrace();

}

return tableList;

}

```

```

private void DisplayTable(){

ArrayList<SetterIT> list = tableList();

DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

Object [] row = new Object [4];

for(int i=0; i<list.size(); i++){

    row[0]= list.get(i).getAT();

    row[1]= list.get(i).getPassword();

    row[2]= list.get(i).getStart();

    if(list.get(i).getEnd()!= null){

        row[3]= list.get(i).getEnd();

    }

    else{

        row[3]= "NULL";

    }

    model.addRow(row);

}

}

```

```

public void executeQuery(String query, String message){

    Connection con = (new GetConnection()).getConnection();

    Statement st;

    try{

        st = con.createStatement();

        if((st.executeUpdate(query))==1){

            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

            model.setRowCount(0);

            DisplayTable();

        }

    }
}

```

```

        JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");

    }else{

        JOptionPane.showMessageDialog(null, "Data Not"+message);

    }

}catch(Exception ex){

    ex.printStackTrace();

}

}

```

```

@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();

```

```

    jLabel4 = new javax.swing.JLabel();

```

```

    jTextField1 = new javax.swing.JTextField();

```

```

    jTextField2 = new javax.swing.JTextField();

```

```

    jTextField3 = new javax.swing.JTextField();

```

```

    jTextField4 = new javax.swing.JTextField();

```

```

    jButton1 = new javax.swing.JButton();

```

```

    jButton2 = new javax.swing.JButton();

```

```

    jButton3 = new javax.swing.JButton();

```

```

    jScrollPane1 = new javax.swing.JScrollPane();

```

```

    jTable1 = new javax.swing.JTable();

```

```

    jButton4 = new javax.swing.JButton();

```

```

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

```

```

    jLabel1.setText("IT_AT");

```

```

    jLabel2.setText("password");

```

```

    jLabel3.setText("start_date");

```

```

    jLabel4.setText("end_date");

```

```

jButton1.setText("INSERT");

jButton1.addActionListener(new java.awt.event.ActionListener() {

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

        jButton1ActionPerformed(evt);

    }

});

```

```

jButton2.setText("UPDATE");

jButton2.addActionListener(new java.awt.event.ActionListener() {

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

        jButton2ActionPerformed(evt);

    }

});

```

```

jButton3.setText("DELETE");

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

        jButton3ActionPerformed(evt);

    }

});

```

```

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "IT_AT", "password", "start_date", "end_date"

    }

) {

    boolean[] canEdit = new boolean [] {

        false, false, true, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {

```

```

public void mouseClicked(java.awt.event.MouseEvent evt) {
    jTable1MouseClicked(evt);
}

});

jScrollPane1.setViewportView(jTable1);

jButton4.setText("Menu");

jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton4ActionPerformed(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()
            .addContainerGap()
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(18, 18, 18)
                    .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 63, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(18, 18, 18)
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                        .addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT_SIZE, 129, Short.MAX_VALUE)
                        .addComponent(jTextField2)
                        .addComponent(jTextField3)
                        .addComponent(jTextField4))
                    .addGap(18, 18, 18))
                .addGroup(layout.createSequentialGroup()
                    .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, 63, Short.MAX_VALUE)
                    .addGap(18, 18, 18)))
            .addContainerGap(18, true))
        .addGroup(layout.createSequentialGroup()
            .addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE, 63, Short.MAX_VALUE)
            .addContainerGap())
    );

```

```

.addGroup(layout.createSequentialGroup())

.addContainerGap()

.addComponent(jButton1)

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

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

    .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

    .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

.addComponent(jButton3)))

.addGap(43, 43, 43)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 452, 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()

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

            .addGroup(layout.createSequentialGroup()

                .addGap(30, 30, 30)

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

                    .addComponent(jLabel1)

                    .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                    .addComponent(jLabel2)

                    .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                    .addComponent(jLabel3)

                    .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                    .addComponent(jLabel4)

                    .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

                .addGap(49, 49, 49)

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

                    .addComponent(jButton1)

                    .addComponent(jButton2)

                    .addComponent(jButton3))

```

```

        .addGap(18, 18, 18)

        .addComponent(jButton4))

    .addGroup(layout.createSequentialGroup())

    .addContainerGap()

    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 275, javax.swing.GroupLayout.PREFERRED_SIZE)))

    .addContainerGap(19, Short.MAX_VALUE))

};

pack();

} // </editor-fold> // GEN-END: initComponents

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) // GEN-FIRST:event_jTable1MouseClicked

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

    int selectedRowIndex = jTable1.getSelectedRow();

    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());

    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());

    jTextField3.setText(model.getValueAt(selectedRowIndex, 2).toString());

    jTextField4.setText(model.getValueAt(selectedRowIndex, 3).toString());

} // GEN-LAST:event_jTable1MouseClicked

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton1ActionPerformed

    String query;

    if(jTextField4.getText().equals("NULL")){

        query = "INSERT INTO IT VALUES ( '"+jTextField1.getText()+"', DEFAULT, '"+jTextField3.getText()+"', NULL );";

    }else{

        query = "INSERT INTO IT VALUES ( '"+jTextField1.getText()+"', DEFAULT, '"+jTextField3.getText()+"', '"+jTextField4.getText()+"' );";

    }

    executeQuery(query, "Inserted");

} // GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton2ActionPerformed

    String query;

    if(jTextField4.getText().equals("NULL")){

        query = "UPDATE IT SET password= '"+jTextField2.getText()+"', start_date= '"+jTextField3.getText()+"', end_date=NULL WHERE IT_AT= '"+jTextField1.getText()+"' ";

    }else{

        query = "UPDATE IT SET password= '"+jTextField2.getText()+"', start_date= '"+jTextField3.getText()+"', end_date= '"+jTextField4.getText()+"' WHERE IT_AT= '"+jTextField1.getText()+"' ";

    }

}

```

```

        executeQuery(query, "Updated");
    } //GEN-LAST:event_jButton2ActionPerformed

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton3ActionPerformed
    String query = "DELETE FROM IT WHERE IT_AT = '"+jTextField1.getText()+"' ";
    executeQuery(query, "Delete");
} //GEN-LAST:event_jButton3ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton4ActionPerformed
    this.setVisible(false);
    new Menu().setVisible(true);
} //GEN-LAST:event_jButton4ActionPerformed

public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new IT().setVisible(true);
        }
    });
}

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField4;
// End of variables declaration//GEN-END:variables
}

```

```
package bd;
```

```
import java.sql.*;
```

```
import java.util.ArrayList;
```

```
import javax.swing.*;
```

```
import javax.swing.table.*;
```

```
public class Language extends javax.swing.JFrame {
```

```
    public Language() {
```

```
        initComponents();
```

```
        jTextField3.setVisible(false);
```

```
        this.setLocationRelativeTo(null);
```

```
        DisplayTable();
```

```
    }
```

```
    public ArrayList<SetterLanguage> tableList(){
```

```
        ArrayList<SetterLanguage> tableList = new ArrayList<>();
```

```
        Connection con = (new GetConnection()).getConnection();
```

```
        String query = "select * from language ";
```

```
        Statement st;
```

```
        ResultSet rs;
```

```
        try{
```

```
            st = con.createStatement();
```

```
            rs = st.executeQuery(query);
```

```
            SetterLanguage setter;
```

```
            while(rs.next()){
```

```
                setter = new SetterLanguage(rs.getString("Ing_gui_AT"),rs.getString("Ing_language"));
```

```
                tableList.add(setter);
```

```

    }

}catch(Exception ex){

    ex.printStackTrace();

}

return tableList;

}

private void DisplayTable(){

ArrayList<SetterLanguage> list = tableList();

DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

Object [] row = new Object [2];

for(int i=0; i<list.size(); i++){

    row[0]= list.get(i).getAT();

    row[1]= list.get(i).getLanguage();

    model.addRow(row);

}

}

public void executeQuery(String query, String message){

    Connection con = (new GetConnection()).getConnection();

    Statement st;

    try{

        st = con.createStatement();

        if((st.executeUpdate(query))==1){

            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

            model.setRowCount(0);

            DisplayTable();

            JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");

        }else{

            JOptionPane.showMessageDialog(null, "Data Not"+message);

        }

    }catch(Exception ex){

        ex.printStackTrace();

    }

}

}

```

```

@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();

    jTextField1 = new javax.swing.JTextField();

    jTextField2 = new javax.swing.JTextField();

    jButton1 = new javax.swing.JButton();

    jButton2 = new javax.swing.JButton();

    jButton3 = new javax.swing.JButton();

    jScrollPane1 = new javax.swing.JScrollPane();

    jTable1 = new javax.swing.JTable();

    jButton4 = new javax.swing.JButton();

    jTextField3 = new javax.swing.JTextField();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel1.setText("Ing_gui_AT");

    jLabel2.setText("Ing_language");

    jButton1.setText("INSERT");

    jButton1.addActionListener(new java.awt.event.ActionListener() {

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

            jButton1ActionPerformed(evt);

        }

    });

    jButton2.setText("UPDATE");

    jButton2.addActionListener(new java.awt.event.ActionListener() {

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

            jButton2ActionPerformed(evt);

        }

    });

    jButton3.setText("DELETE");

```

```

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

        jButton3ActionPerformed(evt);

    }

});

```

```

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "lng_gui_AT", "lng_language"

    }

) {

    boolean[] canEdit = new boolean [] {

        true, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

```

```

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {

    public void mouseClicked(java.awt.event.MouseEvent evt) {

        jTable1MouseClicked(evt);

    }

});

```

```

jScrollPane1.setViewportViewView(jTable1);

```

```

jButton4.setText("Menu");

jButton4.addActionListener(new java.awt.event.ActionListener() {

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

        jButton4ActionPerformed(evt);

    }

});

```

```

jTextField3.setText("jTextField3");

```

```

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

```

```

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

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

    .addGroup(layout.createSequentialGroup()

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

            .addGroup(layout.createSequentialGroup()

                .addGap(34, 34, 34)

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

                    .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, 84, Short.MAX_VALUE)

                    .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

                .addGap(18, 18, 18)

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

                    .addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT_SIZE, 98, Short.MAX_VALUE)

                    .addComponent(jTextField2)))

            .addGroup(layout.createSequentialGroup()

                .addContainerGap()

                .addComponent(jButton1)

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

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

                    .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                    .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

                .addGap(11, 11, 11)

                .addComponent(jButton3))

            .addGroup(layout.createSequentialGroup()

                .addGap(33, 33, 33)

                .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, 99, javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 33, Short.MAX_VALUE)

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 352, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGap(18, 18, 18))

    );

layout.setVerticalGroup(

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

    .addGroup(layout.createSequentialGroup()

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

            .addGroup(layout.createSequentialGroup()

                .addGap(42, 42, 42)

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

                    .addComponent(jLabel1)

                    .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

                .addGap(21, 21, 21)

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

```

```

        .addComponent(jLabel2)

        .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGap(42, 42, 42)

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

            .addComponent(jButton1)

            .addComponent(jButton3)

            .addComponent(jButton2))

        .addGap(18, 18, 18)

        .addComponent(jButton4)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

        .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGap(55, 55, 55))

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

            .addContainerGap(20, Short.MAX_VALUE)

            .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 312, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addGap(16, 16, 16))

    );

    pack();
}

// </editor-fold> //GEN-END: initComponents

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) //GEN-FIRST:event_jTable1MouseClicked

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

    int selectedRowIndex = jTable1.getSelectedRowIndex();

    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());

    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());

    jTextField3.setText(model.getValueAt(selectedRowIndex, 1).toString());

}

//GEN-LAST:event_jTable1MouseClicked

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton1ActionPerformed

    String query = "INSERT INTO language VALUES ( '"+jTextField1.getText()+"', '"+jTextField2.getText()+"' );";

    executeQuery(query, "Inserted");

}

//GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton2ActionPerformed

    String query = "UPDATE language SET lng_language= '"+jTextField2.getText()+"' WHERE lng_gui_AT= '"+jTextField1.getText()+"' and lng_language= '"+jTextField3.getText()+"'";

    executeQuery(query, "Updated");

```

```

} //GEN-LAST:event_jButton2ActionPerformed

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton3ActionPerformed
    String query = "DELETE FROM language WHERE lng_gui_AT= '"+jTextField1.getText()+"' and lng_language= '"+jTextField3.getText()+"'";
    executeQuery(query, "Delete");
} //GEN-LAST:event_jButton3ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton4ActionPerformed
    this.setVisible(false);
    new Menu().setVisible(true); } //GEN-LAST:event_jButton4ActionPerformed

public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Language().setVisible(true);
        }
    });
}

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
// End of variables declaration//GEN-END:variables
}

```

```

package bd;

import java.lang.System.Logger;

import java.lang.System.Logger.Level;

import java.sql.*;

import javax.swing.JOptionPane;

import java.util.logging.*;

public class LoginForm extends javax.swing.JFrame {

    public LoginForm() {

        initComponents();

        this.setLocationRelativeTo(null);

    }

    @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();

        jTextField1 = new javax.swing.JTextField();

        jPasswordField1 = new javax.swing.JPasswordField();

        jButton1 = new javax.swing.JButton();

        jLabel3 = new javax.swing.JLabel();

        jLabel4 = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        setLocation(new java.awt.Point(500, 200));

        jLabel1.setText("username");

        jLabel2.setText("password");

        jPasswordField1.addActionListener(new java.awt.event.ActionListener() {

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

```



```

        .addComponent(jLabel4)

        .addGap(85, 85, 85))

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

    .addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED_SIZE, 129, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addGap(188, 188, 188))))

);

layout.setVerticalGroup(

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

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

    .addGap(10, 10, 10)

    .addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED_SIZE, 25, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addGap(18, 18, 18)

    .addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED_SIZE, 30, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 27, Short.MAX_VALUE)

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

        .addComponent(jLabel1)

        .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

    .addGap(35, 35, 35)

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

        .addComponent(jLabel2)

        .addComponent(jPasswordField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

    .addGap(48, 48, 48)

    .addComponent(jButton1)

    .addGap(102, 102, 102))

);

pack();

}

// </editor-fold> //GEN-END: initComponents

private void jPasswordField1ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jPasswordField1ActionPerformed

    // TODO add your handling code here:

}

//GEN-LAST:event_jPasswordField1ActionPerformed

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton1ActionPerformed

    try{

        String query1 = "update IT set active='INACTIVE';" ;

        PreparedStatement preparedS1 = (new GetConnection()).getConnection().prepareStatement(query1);

        preparedS1.executeUpdate(query1);

```

```

        PreparedStatement ps = (new GetConnection()).getConnection().prepareStatement("Select wrk_AT, wrk_Iname, password from worker inner join IT on wrk_AT=IT_AT where wrk_Iname=? and password=?");

```

```

        ps.setString(1,jTextField1.getText());

```

```

        ps.setString(2,String.valueOf(jPasswordField1.getPassword()));

```

```

        ResultSet rs = ps.executeQuery();

```

```

        if(rs.next()){

```

```

            String it_at = rs.getString("wrk_AT");

```

```

            String query2 = "update IT set active='ACTIVE' where IT_AT='"+it_at+"' ;" ;

```

```

            PreparedStatement preparedS2 = (new GetConnection()).getConnection().prepareStatement(query2);

```

```

            preparedS2.executeUpdate(query2);

```

```

                this.setVisible(false);

```

```

                new Menu().setVisible(true);

```

```

        }else{

```

```

            JOptionPane.showMessageDialog(this, "Username Or Password Are Invalid");

```

```

        }

```

```

    }catch(SQLException ex){

```

```

        ex.printStackTrace();

```

```

    }

```

```

} //GEN-LAST:event_jButton1ActionPerformed

```

```

public static void main(String args[]) {

```

```

    java.awt.EventQueue.invokeLater(new Runnable() {

```

```

        public void run() {

```

```

            new LoginForm().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.JLabel jLabel4;

```

```

private javax.swing.JPasswordField jPasswordField1;

```

```
private javax.swing.JTextField jTextField1;

// End of variables declaration//GEN-END:variables

}
```

```
package bd;
```

```
import java.sql.*;
import java.util.ArrayList;
import javax.swing.*;
import javax.swing.table.*;
```

```
public class Manages extends javax.swing.JFrame {
```

```
    public Manages() {
        initComponents();
        this.setLocationRelativeTo(null);
        DisplayTable();
    }
```

```
    public ArrayList<SetterManages> tableList(){
```

```
        ArrayList<SetterManages> tableList = new ArrayList<>();
        Connection con = (new GetConnection()).getConnection();
        String query = "select * from manages ";
        Statement st;
        ResultSet rs;
```

```
        try{
            st = con.createStatement();
            rs = st.executeQuery(query);
            SetterManages setter;
            while(rs.next()){
```

```

        setter = new SetterManages(rs.getString("mng_adm_AT"), rs.getInt("mng_br_code"));

        tableList.add(setter);

    }

} catch (Exception ex) {

    ex.printStackTrace();

}

return tableList;

}

```

```

private void DisplayTable(){

    ArrayList<SetterManages> list = tableList();

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

    Object [] row = new Object [2];

    for(int i=0; i<list.size(); i++){

        row[0]= list.get(i).getAT();

        row[1]= list.get(i).getCode();

        model.addRow(row);

    }

}

```

```

public void executeQuery(String query, String message){

    Connection con = (new GetConnection()).getConnection();

    Statement st;

    try{

        st = con.createStatement();

        if((st.executeUpdate(query))==1){

            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

            model.setRowCount(0);

            DisplayTable();

            JOptionPane.showMessageDialog(null, "Data "+message+"Sucesfully");

        }else{

            JOptionPane.showMessageDialog(null, "Data Not"+message);

        }

    } catch (Exception ex){

        ex.printStackTrace();

    }

}

```

```
}
```

```
@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();
```

```
    jTextField1 = new javax.swing.JTextField();
```

```
    jTextField2 = new javax.swing.JTextField();
```

```
    jButton1 = new javax.swing.JButton();
```

```
    jButton2 = new javax.swing.JButton();
```

```
    jButton3 = new javax.swing.JButton();
```

```
    jScrollPane1 = new javax.swing.JScrollPane();
```

```
    jTable1 = new javax.swing.JTable();
```

```
    jButton4 = new javax.swing.JButton();
```

```
    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
```

```
    jLabel1.setText("mng_adm_AT");
```

```
    jLabel2.setText("mng_br_code");
```

```
    jButton1.setText("INSERT");
```

```
    jButton1.addActionListener(new java.awt.event.ActionListener() {
```

```
        public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
            jButton1ActionPerformed(evt);
```

```
        }
```

```
    });
```

```
    jButton2.setText("UPDATE");
```

```
    jButton2.addActionListener(new java.awt.event.ActionListener() {
```

```
        public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
            jButton2ActionPerformed(evt);
```

```
        }
```

```
    });
```

```

jButton3.setText("DELETE");

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

        jButton3ActionPerformed(evt);

    }

});

```

```

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "mng_adm_AT", "mng_br_code"

    }

) {

    boolean[] canEdit = new boolean [] {

        false, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

```

```

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {

    public void mouseClicked(java.awt.event.MouseEvent evt) {

        jTable1MouseClicked(evt);

    }

});

jScrollPane1.setViewportView(jTable1);

```

```

jButton4.setText("Menu");

jButton4.addActionListener(new java.awt.event.ActionListener() {

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

        jButton4ActionPerformed(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())

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

    .addGroup(layout.createSequentialGroup())

    .addGap(26, 26, 26)

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

    .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

    .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, 80, Short.MAX_VALUE))

    .addGap(18, 18, 18)

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

    .addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT_SIZE, 110, Short.MAX_VALUE)

    .addComponent(jTextField2)))

    .addGroup(layout.createSequentialGroup())

    .addContainerGap()

    .addComponent(jButton1)

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

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

    .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

    .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

    .addComponent(jButton3)))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 47, Short.MAX_VALUE)

    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 292, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addGap(37, 37, 37))

);

layout.setVerticalGroup(

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

    .addGroup(layout.createSequentialGroup())

    .addGap(54, 54, 54)

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

    .addComponent(jLabel1)

    .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(jLabel2)

    .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

    .addGap(50, 50, 50)

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

```



```

        .addComponent(jButton1)

        .addComponent(jButton2)

        .addComponent(jButton3))

    .addGap(18, 18, 18)

    .addComponent(jButton4)

    .addContainerGap(96, Short.MAX_VALUE))

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

        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 281, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGap(21, 21, 21))

    );

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

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) { // GEN-FIRST: event_jTable1MouseClicked

    DefaultTableModel model = (DefaultTableModel) jTable1.getModel();

    int selectedRowIndex = jTable1.getSelectedRow();

    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());

    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());
} // GEN-LAST: event_jTable1MouseClicked

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) { // GEN-FIRST: event_jButton1ActionPerformed

    String query = "INSERT INTO manages VALUES ('"+jTextField1.getText()+"','"+jTextField2.getText()+"')";

    executeQuery(query, "Inserted");
} // GEN-LAST: event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) { // GEN-FIRST: event_jButton2ActionPerformed

    String query = "UPDATE manages SET mng_br_code='"+jTextField2.getText()+" WHERE mng_adm_AT='"+jTextField1.getText()+"'";

    executeQuery(query, "Updated");

    //den exei noima na kano upadate me vasi to br_code giati

    //tha allazoun ta am olon oson einai sthn etairia ayth
} // GEN-LAST: event_jButton2ActionPerformed

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) { // GEN-FIRST: event_jButton3ActionPerformed

    String query = "DELETE FROM manages WHERE mng_adm_AT='"+jTextField1.getText()+"'";

    executeQuery(query, "Delete");
} // GEN-LAST: event_jButton3ActionPerformed

```

```
}//GEN-LAST:event_jButton3ActionPerformed
```

```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_jButton4ActionPerformed
```

```
    this.setVisible(false);
```

```
    new Menu().setVisible(true); }//GEN-LAST:event_jButton4ActionPerformed
```

```
public static void main(String args[]) {
```

```
    java.awt.EventQueue.invokeLater(new Runnable() {
```

```
        public void run() {
```

```
            new Manages().setVisible(true);
```

```
        }
```

```
    });
```

```
}
```

```
// Variables declaration - do not modify//GEN-BEGIN:variables
```

```
private javax.swing.JButton jButton1;
```

```
private javax.swing.JButton jButton2;
```

```
private javax.swing.JButton jButton3;
```

```
private javax.swing.JButton jButton4;
```

```
private javax.swing.JLabel jLabel1;
```

```
private javax.swing.JLabel jLabel2;
```

```
private javax.swing.JScrollPane jScrollPane1;
```

```
private javax.swing.JTable jTable1;
```

```
private javax.swing.JTextField jTextField1;
```

```
private javax.swing.JTextField jTextField2;
```

```
// End of variables declaration//GEN-END:variables
```

```
}
```

```
package bd;
```

```
public class Menu extends javax.swing.JFrame {
```

```

public Menu() {

    initComponents();

    this.setLocationRelativeTo(null);
}

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-BEGIN: initComponents
private void initComponents() {

    javax.swing.ButtonGroup buttonGroup1 = new javax.swing.ButtonGroup();

    jRadioButton1 = new javax.swing.JRadioButton();

    jRadioButton2 = new javax.swing.JRadioButton();

    jRadioButton3 = new javax.swing.JRadioButton();

    jRadioButton4 = new javax.swing.JRadioButton();

    jRadioButton5 = new javax.swing.JRadioButton();

    jRadioButton6 = new javax.swing.JRadioButton();

    jButton1 = new javax.swing.JButton();

    jRadioButton7 = new javax.swing.JRadioButton();

    jButton2 = new javax.swing.JButton();

    jLabel1 = new javax.swing.JLabel();

    jLabel2 = new javax.swing.JLabel();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    setBackground(new java.awt.Color(204, 204, 255));

    setLocation(new java.awt.Point(500, 200));

    buttonGroup1.add(jRadioButton1);

    jRadioButton1.setText("INSERT/UPDATE/DELETE");

    jRadioButton1.setPreferredSize(new java.awt.Dimension(155, 25));

    jRadioButton1.addActionListener(new java.awt.event.ActionListener() {

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

            jRadioButton1ActionPerformed(evt);

        }

    });

    buttonGroup1.add(jRadioButton2);

    jRadioButton2.setText("Search Trip");

    jRadioButton2.setPreferredSize(new java.awt.Dimension(155, 25));

```

```
jRadioButton2.addActionListener(new java.awt.event.ActionListener() {  
  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
  
        jRadioButton2ActionPerformed(evt);  
  
    }  
  
});
```

```
buttonGroup1.add(jRadioButton3);  
  
jRadioButton3.setText("Customer's Reservation");  
  
jRadioButton3.setPreferredSize(new java.awt.Dimension(155, 25));  
  
jRadioButton3.addActionListener(new java.awt.event.ActionListener() {  
  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
  
        jRadioButton3ActionPerformed(evt);  
  
    }  
  
});
```

```
buttonGroup1.add(jRadioButton4);  
  
jRadioButton4.setText("Branch's Infos");  
  
jRadioButton4.setPreferredSize(new java.awt.Dimension(155, 25));  
  
jRadioButton4.addActionListener(new java.awt.event.ActionListener() {  
  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
  
        jRadioButton4ActionPerformed(evt);  
  
    }  
  
});
```

```
buttonGroup1.add(jRadioButton5);  
  
jRadioButton5.setText("Branch's Workers");  
  
jRadioButton5.setPreferredSize(new java.awt.Dimension(155, 25));  
  
jRadioButton5.addActionListener(new java.awt.event.ActionListener() {  
  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
  
        jRadioButton5ActionPerformed(evt);  
  
    }  
  
});
```

```
buttonGroup1.add(jRadioButton6);  
  
jRadioButton6.setText("Show Log");  
  
jRadioButton6.setPreferredSize(new java.awt.Dimension(155, 25));  
  
jRadioButton6.addActionListener(new java.awt.event.ActionListener() {  
  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
  
        jRadioButton6ActionPerformed(evt);  
  
    }  
  
});
```

```

    }

});

jButton1.setText("SELECT");

jButton1.addActionListener(new java.awt.event.ActionListener() {

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

        jButton1ActionPerformed(evt);

    }

});

buttonGroup1.add(jRadioButton7);

jRadioButton7.setText("Revenue and Cost");

jRadioButton7.setPreferredSize(new java.awt.Dimension(155, 25));

jButton2.setText("LOG OUT");

jButton2.addActionListener(new java.awt.event.ActionListener() {

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

        jButton2ActionPerformed(evt);

    }

});

jLabel1.setFont(new java.awt.Font("Segoe UI", 3, 18)); // NOI18N
jLabel1.setText("    MENU");

jLabel2.setText("Choose an option to continue");

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

layout.setHorizontalGroup(

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

        .addGroup(layout.createSequentialGroup()

            .addGap(80, 80, 80)

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

                .addComponent(jRadioButton3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                .addComponent(jRadioButton4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                .addComponent(jRadioButton5, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

            )

            .addGap(103, 103, 103)

        )

);

```

```

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

.addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addComponent(jButton1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

.addGap(0, 112, Short.MAX_VALUE))

.addGroup(layout.createSequentialGroup())

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

.addComponent(jRadioButton6, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(jRadioButton2, javax.swing.GroupLayout.PREFERRED_SIZE, 155, javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(jRadioButton7, javax.swing.GroupLayout.PREFERRED_SIZE, 137, javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(jRadioButton1, javax.swing.GroupLayout.PREFERRED_SIZE, 185, javax.swing.GroupLayout.PREFERRED_SIZE))

.addContainerGap(264, Short.MAX_VALUE)))

.addGroup(layout.createSequentialGroup())

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

.addGroup(layout.createSequentialGroup())

.addGap(201, 201, 201)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 125, javax.swing.GroupLayout.PREFERRED_SIZE))

.addGroup(layout.createSequentialGroup())

.addGap(183, 183, 183)

.addComponent(jLabel2)))

.addGap(0, 0, Short.MAX_VALUE))

);

layout.setVerticalGroup(

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

.addGroup(layout.createSequentialGroup())

.addGap(14, 14, 14)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 30, javax.swing.GroupLayout.PREFERRED_SIZE)

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

.addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED_SIZE, 28, javax.swing.GroupLayout.PREFERRED_SIZE)

.addGap(18, 18, 18)

.addComponent(jRadioButton1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

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

.addComponent(jRadioButton2, javax.swing.GroupLayout.PREFERRED_SIZE, 25, javax.swing.GroupLayout.PREFERRED_SIZE)

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

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

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

.addComponent(jButton1)

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

.addComponent(jButton2))

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

```

```

        .addComponent(jRadioButton3, javax.swing.GroupLayout.PREFERRED_SIZE, 25, javax.swing.GroupLayout.PREFERRED_SIZE)

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

        .addComponent(jRadioButton4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

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

        .addComponent(jRadioButton5, javax.swing.GroupLayout.PREFERRED_SIZE, 25, javax.swing.GroupLayout.PREFERRED_SIZE)

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

        .addComponent(jRadioButton6, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

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

        .addComponent(jRadioButton7, javax.swing.GroupLayout.PREFERRED_SIZE, 21, javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addContainerGap(102, Short.MAX_VALUE))

    );

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

private void jRadioButton1ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jRadioButton1ActionPerformed

    // TODO add your handling code here:

} //GEN-LAST:event_jRadioButton1ActionPerformed

private void jRadioButton3ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jRadioButton3ActionPerformed

    // TODO add your handling code here:

} //GEN-LAST:event_jRadioButton3ActionPerformed

private void jRadioButton2ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jRadioButton2ActionPerformed

    // TODO add your handling code here:

} //GEN-LAST:event_jRadioButton2ActionPerformed

private void jRadioButton6ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jRadioButton6ActionPerformed

    // TODO add your handling code here:

} //GEN-LAST:event_jRadioButton6ActionPerformed

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton1ActionPerformed

    if(jRadioButton1.isSelected()==true){

        this.setVisible(false);

        new ChooseTable().setVisible(true);

    }

    if(jRadioButton2.isSelected()==true){

        this.setVisible(false);

        new SearchTrip().setVisible(true);

    }

}

```

```

    }

    if(jRadioButton3.isSelected()==true){

        this.setVisible(false);

        new CustomerReservation().setVisible(true);

    }

    if(jRadioButton4.isSelected()==true){

        this.setVisible(false);

        new BranchInfo().setVisible(true);

    }

    if(jRadioButton5.isSelected()==true){

        this.setVisible(false);

        new BranchWorker().setVisible(true);

    }

    if(jRadioButton6.isSelected()==true){

        this.setVisible(false);

        new ShowLog().setVisible(true);

    }

    if(jRadioButton7.isSelected()==true){

        this.setVisible(false);

        new RevenueAndCost().setVisible(true);

    }

}

//GEN-LAST:event_jButton1ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton4ActionPerformed
    // TODO add your handling code here:
}

//GEN-LAST:event_jButton4ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton2ActionPerformed
    this.setVisible(false);

    new LoginForm().setVisible(true);
}

//GEN-LAST:event_jButton2ActionPerformed

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton5ActionPerformed
    // TODO add your handling code here:
}

//GEN-LAST:event_jButton5ActionPerformed

public static void main(String args[]) {

```



```

java.awt.EventQueue.invokeLater(new Runnable() {

    public void run() {

        new Menu().setVisible(true);

    }

});
}

```

```

// Variables declaration - do not modify//GEN-BEGIN:variables

```

```

private javax.swing.JButton jButton1;

```

```

private javax.swing.JButton jButton2;

```

```

private javax.swing.JLabel jLabel1;

```

```

private javax.swing.JLabel jLabel2;

```

```

private javax.swing.JRadioButton jRadioButton1;

```

```

private javax.swing.JRadioButton jRadioButton2;

```

```

private javax.swing.JRadioButton jRadioButton3;

```

```

private javax.swing.JRadioButton jRadioButton4;

```

```

private javax.swing.JRadioButton jRadioButton5;

```

```

private javax.swing.JRadioButton jRadioButton6;

```

```

private javax.swing.JRadioButton jRadioButton7;

```

```

// End of variables declaration//GEN-END:variables

```

```

}

```

```

-----

```

```

package bd;

```

```

import java.sql.*;

```

```

import java.util.ArrayList;

```

```

import javax.swing.*;

```

```

import javax.swing.table.*;

```

```

public class Offers extends javax.swing.JFrame {

```

```

    public Offers() {

```

```

        initComponents();

```

```

this.setLocationRelativeTo(null);

DisplayTable();

}

```

```

public ArrayList<SetterOffers> tableList(){

    ArrayList<SetterOffers> tableList = new ArrayList<>();

    Connection con = (new GetConnection()).getConnection();

    String query = "select * from offers ";

    Statement st;

    ResultSet rs;

    try{

        st = con.createStatement();

        rs = st.executeQuery(query);

        SetterOffers setter;

        while(rs.next()){

            setter = new SetterOffers(rs.getInt("off_id"),rs.getString("off_start"), rs.getString("off_end"), rs.getInt("off_cost"), rs.getInt("off_dst_id"));

            tableList.add(setter);

        }

    }catch(Exception ex){

        ex.printStackTrace();

    }

    return tableList;

}

```

```

private void DisplayTable(){

    ArrayList<SetterOffers> list = tableList();

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

    Object [] row = new Object [5];

    for(int i=0; i<list.size(); i++){

        row[0]= list.get(i).getID();

        row[1]= list.get(i).getStart();

        row[2]= list.get(i).getEnd();

        row[3]= list.get(i).getCost();

        row[4]= list.get(i).getDst_id();

    }
}

```

```

        model.addRow(row);
    }
}

public void executeQuery(String query, String message){
    Connection con = (new GetConnection()).getConnection();
    Statement st;

    try{
        st = con.createStatement();
        if((st.executeUpdate(query))!=1){
            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();
            model.setRowCount(0);
            DisplayTable();
            JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");
        }else{
            JOptionPane.showMessageDialog(null, "Data Not"+message);
        }
    }catch(Exception ex){
        ex.printStackTrace();
    }

}

@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();
    jLabel4 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    jTextField1 = new javax.swing.JTextField();
    jTextField2 = new javax.swing.JTextField();
    jTextField3 = new javax.swing.JTextField();
    jTextField4 = new javax.swing.JTextField();
    jTextField5 = new javax.swing.JTextField();

```

```

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

jButton3 = new javax.swing.JButton();

jScrollPane1 = new javax.swing.JScrollPane();

jTable1 = new javax.swing.JTable();

jButton4 = new javax.swing.JButton();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);


jLabel1.setText("off_id");


jLabel2.setText("off_start");


jLabel3.setText("off_end");


jLabel4.setText("off_cost");


jLabel5.setText("off_dst_id");


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


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


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

```

```
jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        ,
        new String [] {
            "off_id", "off_start", "off_end", "off_cost", "off_dst_id"
        }
    }
) {
    boolean[] canEdit = new boolean [] {
        false, false, false, false, false
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return canEdit [columnIndex];
    }
});

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        jTable1MouseClicked(evt);
    }
});

jScrollPane1.setViewportView(jTable1);

 jButton4.setText("Menu");

 jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton4ActionPerformed(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)
                .add(layout.createSequentialGroup()
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addGroup(layout.createSequentialGroup())
                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                            .addGroup(layout.createSequentialGroup())
                            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```

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

.addContainerGap()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED_SIZE, 59, javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED_SIZE, 59, javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED_SIZE, 59, javax.swing.GroupLayout.PREFERRED_SIZE))

.addGap(18, 18, 18))

.addGroup(layout.createSequentialGroup())

.addGap(27, 27, 27)

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

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 37, javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED_SIZE, 59, javax.swing.GroupLayout.PREFERRED_SIZE))

.addGap(18, 18, 18)))

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

.addComponent(jTextField4, javax.swing.GroupLayout.DEFAULT_SIZE, 145, Short.MAX_VALUE)

.addComponent(jTextField5)

.addComponent(jTextField2)

.addComponent(jTextField3)

.addComponent(jTextField1)))

.addGroup(layout.createSequentialGroup())

.addContainerGap()

.addComponent(jButton1)

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

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

.addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

.addComponent(jButton3)))

.addGap(27, 27, 27)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 804, javax.swing.GroupLayout.PREFERRED_SIZE)

.addGap(16, 16, 16))

);

layout.setVerticalGroup(

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

.addGroup(layout.createSequentialGroup())

.addGap(25, 25, 25)

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

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 396, javax.swing.GroupLayout.PREFERRED_SIZE)

.addGroup(layout.createSequentialGroup())

```

```

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

    .addComponent(jLabel1)

    .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

    .addComponent(jLabel2)

    .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

    .addComponent(jLabel3)

    .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

    .addComponent(jLabel4)

    .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

    .addComponent(jLabel5)

    .addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

.addGap(34, 34, 34)

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

    .addComponent(jButton1)

    .addComponent(jButton2)

    .addComponent(jButton3))

.addGap(18, 18, 18)

.addComponent(jButton4)))

.addContainerGap(39, Short.MAX_VALUE)

);

pack();
}

// </editor-fold> //GEN-END: initComponents

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) { //GEN-FIRST:event_jTable1MouseClicked

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

    int selectedRowIndex = jTable1.getSelectedRow();

    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());

    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());

```

```

        jTextField3.setText(model.getValueAt(selectedRowIndex, 2).toString());

        jTextField4.setText(model.getValueAt(selectedRowIndex, 3).toString());

        jTextField5.setText(model.getValueAt(selectedRowIndex, 4).toString());

    } //GEN-LAST:event_jTable1MouseClicked

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton1ActionPerformed

    String query = "INSERT INTO offers VALUES ( NULL, '"+jTextField2.getText()+"', '"+jTextField3.getText()+"', '"+jTextField4.getText()+"', '"+jTextField5.getText()+"' );";

    executeQuery(query, "Inserted");

} //GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton2ActionPerformed

    String query = "UPDATE offers SET off_start= '"+jTextField2.getText()+"', off_end= '"+jTextField3.getText()+"', off_cost= '"+jTextField4.getText()+"', off_dst_id= '"+jTextField5.getText()+"' WHERE off_id= '"+jTextField1.getText()+"' ";

    executeQuery(query, "Updated");

} //GEN-LAST:event_jButton2ActionPerformed

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton3ActionPerformed

    String query = "DELETE FROM offers WHERE off_id = '"+jTextField1.getText()+"' ";

    executeQuery(query, "Delete");

} //GEN-LAST:event_jButton3ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton4ActionPerformed

    this.setVisible(false);

    new Menu().setVisible(true);

} //GEN-LAST:event_jButton4ActionPerformed

public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new Offers().setVisible(true);

        }

    });

}

// Variables declaration - do not modify //GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;

```



```

private javax.swing.JButton jButton4;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTable jTable1;

private javax.swing.JTextField jTextField1;

private javax.swing.JTextField jTextField2;

private javax.swing.JTextField jTextField3;

private javax.swing.JTextField jTextField4;

private javax.swing.JTextField jTextField5;

// End of variables declaration//GEN-END:variables
}

```

```

package bd;

```

```

import java.sql.*;

import java.util.ArrayList;

import javax.swing.*;

import javax.swing.table.*;

```

```

public class Phones extends javax.swing.JFrame {

```

```

    public Phones() {

        initComponents();

        jTextField3.setVisible(false);

        this.setLocationRelativeTo(null);

        DisplayTable();

    }

```

```

public ArrayList<SetterPhones> tableList(){

    ArrayList<SetterPhones> tableList = new ArrayList<>();

    Connection con = (new GetConnection()).getConnection();

    String query = "select * from phones ";

    Statement st;

    ResultSet rs;

    try{

        st = con.createStatement();

        rs = st.executeQuery(query);

        SetterPhones setter;

        while(rs.next()){

            setter = new SetterPhones(rs.getInt("ph_br_code"), rs.getString("ph_number"));

            tableList.add(setter);

        }

    }catch(Exception ex){

        ex.printStackTrace();

    }

    return tableList;

}

```

```

private void DisplayTable(){

    ArrayList<SetterPhones> list = tableList();

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

    Object [] row = new Object [2];

    for(int i=0; i<list.size(); i++){

        row[0]= list.get(i).getCode();

        row[1]= list.get(i).getNumber();

        model.addRow(row);

    }

}

```

```

public void executeQuery(String query, String message){

    Connection con = (new GetConnection()).getConnection();

    Statement st;

```

```

try{

    st = con.createStatement();

    if((st.executeUpdate(query))==1){

        DefaultTableModel model = (DefaultTableModel)Table1.getModel();

        model.setRowCount(0);

        DisplayTable();

        JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");

    }else{

        JOptionPane.showMessageDialog(null, "Data Not"+message);

    }

}catch(Exception ex){

    ex.printStackTrace();

}

}

@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();

    jTextField1 = new javax.swing.JTextField();

    jTextField2 = new javax.swing.JTextField();

    jButton1 = new javax.swing.JButton();

    jButton2 = new javax.swing.JButton();

    jButton3 = new javax.swing.JButton();

    jScrollPane1 = new javax.swing.JScrollPane();

    jTable1 = new javax.swing.JTable();

    jButton4 = new javax.swing.JButton();

    jTextField3 = new javax.swing.JTextField();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel1.setText("ph_br_code");

    jLabel2.setText("ph_number");

    jButton1.setText("INSERT");

```

```

jButton1.addActionListener(new java.awt.event.ActionListener() {

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

        jButton1ActionPerformed(evt);

    }

});

```

```

jButton2.setText("UPDATE");

jButton2.addActionListener(new java.awt.event.ActionListener() {

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

        jButton2ActionPerformed(evt);

    }

});

```

```

jButton3.setText("DELETE");

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

        jButton3ActionPerformed(evt);

    }

});

```

```

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "ph_br_code", "ph_number"

    }

) {

    boolean[] canEdit = new boolean [] {

        true, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {

    public void mouseClicked(java.awt.event.MouseEvent evt) {

        jTable1MouseClicked(evt);

    }

});

```

```

    }

});

jScrollPane1.setViewportViewView(jTable1);


jButton4.setText("Menu");

jButton4.addActionListener(new java.awt.event.ActionListener() {

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

        jButton4ActionPerformed(evt);

    }

});


jTextField3.setText("jTextField3");


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

layout.setHorizontalGroup(

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

        .addGroup(layout.createSequentialGroup()

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

                .addGroup(layout.createSequentialGroup()

                    .addGroup(layout.createSequentialGroup()

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

                            .addGroup(layout.createSequentialGroup()

                                .addGap(46, 46, 46)

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

                                    .addComponent(jLabel1)

                                    .addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED_SIZE, 71, javax.swing.GroupLayout.PREFERRED_SIZE))

                                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

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

                                    .addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT_SIZE, 109, Short.MAX_VALUE)

                                    .addComponent(jTextField2))

                                .addGap(37, 37, 37))

                                .addGroup(layout.createSequentialGroup()

                                    .addContainerGap()

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

                                        .addGroup(layout.createSequentialGroup()

                                            .addGroup(layout.createSequentialGroup()

                                                .addComponent(jButton1)

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

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

                                                    .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                                    .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

```

```

        .addComponent(jButton3))

        .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 21, Short.MAX_VALUE)))

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 302, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addContainerGap())

);

layout.setVerticalGroup(

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

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

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

                .addGroup(layout.createSequentialGroup()

                    .addGap(38, 38, 38)

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

                        .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addComponent(jLabel1))

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

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

                        .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addComponent(jLabel2))

                    .addGap(49, 49, 49)

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

                        .addComponent(jButton1)

                        .addComponent(jButton2)

                        .addComponent(jButton3))

                    .addGap(18, 18, 18)

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

                        .addGroup(layout.createSequentialGroup()

                            .addComponent(jButton4)

                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

                            .addGap(0, 0, Short.MAX_VALUE)

                            .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))))

                .addGroup(layout.createSequentialGroup()

                    .addContainerGap(12, Short.MAX_VALUE)

                    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 286, javax.swing.GroupLayout.PREFERRED_SIZE)))

            .addContainerGap())

);

pack();

```

```
// </editor-fold>//GEN-END:initComponents
```

```
private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {//GEN-FIRST:event_jTable1MouseClicked
```

```
    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();
```

```
    int selectedRowIndex = jTable1.getSelectedRow();
```

```
    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());
```

```
    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());
```

```
    jTextField3.setText(model.getValueAt(selectedRowIndex, 1).toString());
```

```
//GEN-LAST:event_jTable1MouseClicked
```

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_jButton1ActionPerformed
```

```
    String query = "INSERT INTO phones VALUES (" + jTextField1.getText() + "," + jTextField2.getText() + " )";
```

```
    executeQuery(query, "Inserted");
```

```
//GEN-LAST:event_jButton1ActionPerformed
```

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_jButton2ActionPerformed
```

```
    String query = "UPDATE phones SET ph_number= " + jTextField2.getText() + " WHERE ph_number= " + jTextField3.getText() + " ";
```

```
    executeQuery(query, "Updated");
```

```
//GEN-LAST:event_jButton2ActionPerformed
```

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_jButton3ActionPerformed
```

```
    String query = "DELETE FROM phones WHERE ph_number= " + jTextField2.getText() + " ";
```

```
    executeQuery(query, "Delete");
```

```
//GEN-LAST:event_jButton3ActionPerformed
```

```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_jButton4ActionPerformed
```

```
    this.setVisible(false);
```

```
    new Menu().setVisible(true); } //GEN-LAST:event_jButton4ActionPerformed
```

```
public static void main(String args[]) {
```

```
    java.awt.EventQueue.invokeLater(new Runnable() {
```

```
        public void run() {
```

```
            new Phones().setVisible(true);
```

```
        }
```

```
    });
```

```

}

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
// End of variables declaration//GEN-END:variables
}

```

```

package bd;

```

```

import java.sql.*;
import java.util.ArrayList;
import javax.swing.*;
import javax.swing.table.*;

```

```

public class Reservation extends javax.swing.JFrame {

```

```

    public Reservation() {
        initComponents();
        this.setLocationRelativeTo(null);
        DisplayTable();
    }

```



```

public ArrayList<SetterReservation> tableList(){

    ArrayList<SetterReservation> tableList = new ArrayList<>();

    Connection con = (new GetConnection()).getConnection();

    String query = "select * from reservation ";

    Statement st;

    ResultSet rs;

    try{

        st = con.createStatement();

        rs = st.executeQuery(query);

        SetterReservation setter;

        while(rs.next()){

            setter = new SetterReservation(rs.getInt("res_tr_id"), rs.getInt("res_seatnum"),rs.getString("res_name"),rs.getString("res_lname"), rs.getString("res_isadult"));

            tableList.add(setter);

        }

    }catch(Exception ex){

        ex.printStackTrace();

    }

    return tableList;

}

private void DisplayTable(){

    ArrayList<SetterReservation> list = tableList();

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

    Object [] row = new Object [5];

    for(int i=0; i<list.size(); i++){

        row[0]= list.get(i).getID();

        row[1]= list.get(i).getSeatnum();

        row[2]= list.get(i).getName();

        row[3]= list.get(i).getLname();

        row[4]= list.get(i).getIsadult();

        model.addRow(row);

    }

}

```

```

public void executeQuery(String query, String message){

    Connection con = (new GetConnection()).getConnection();

    Statement st;

    try{

        st = con.createStatement();

        if((st.executeUpdate(query))!=1){

            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

            model.setRowCount(0);

            DisplayTable();

            JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");

        }else{

            JOptionPane.showMessageDialog(null, "Data Not"+message);

        }

    }catch(Exception ex){

        ex.printStackTrace();

    }

}

}

@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();

    jLabel4 = new javax.swing.JLabel();

    jLabel5 = new javax.swing.JLabel();

    jTextField1 = new javax.swing.JTextField();

    jTextField2 = new javax.swing.JTextField();

    jTextField3 = new javax.swing.JTextField();

    jTextField4 = new javax.swing.JTextField();

    jTextField5 = new javax.swing.JTextField();

    jButton1 = new javax.swing.JButton();

    jButton2 = new javax.swing.JButton();

    jButton3 = new javax.swing.JButton();

    jScrollPane1 = new javax.swing.JScrollPane();

    jTable1 = new javax.swing.JTable();

```

```

jButton4 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

jLabel1.setText("res_tr_id");

jLabel2.setText("res_seatnum");

jLabel3.setText("res_name");

jLabel4.setText("res_lname");

jLabel5.setText("res_isadult");

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

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

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

jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {

    },

```

```

new String [] {
    "res_tr_id", "res_seatnum", "res_name", "res_lname", "res_isadult"
}
){
    boolean[] canEdit = new boolean [] {
        false, false, false, true, false
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return canEdit [columnIndex];
    }
});

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        jTable1MouseClicked(evt);
    }
});

jScrollPane1.setViewportView(jTable1);

jButton4.setText("Menu");
jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton4ActionPerformed(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(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 60, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .add(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .add(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .add(jLabel4)
                )
            )
        )
    );

```

```

        .addComponent(jLabel5, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

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

        .addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT_SIZE, 127, Short.MAX_VALUE)

        .addComponent(jTextField2)

        .addComponent(jTextField3)

        .addComponent(jTextField4)

        .addComponent(jTextField5)))

    .addGroup(layout.createSequentialGroup()

        .addContainerGap()

        .addComponent(jButton1)

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

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

            .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

            .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

        .addComponent(jButton3)))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 42, Short.MAX_VALUE)

    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 523, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addGap(24, 24, 24))
);

layout.setVerticalGroup(

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

    .addGroup(layout.createSequentialGroup()

        .addGap(29, 29, 29)

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

            .addGroup(layout.createSequentialGroup()

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

                    .addComponent(jLabel1)

                    .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                    .addComponent(jLabel2)

                    .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                    .addComponent(jLabel3)

                    .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

```

```

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

            .addComponent(jLabel4)

            .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

            .addComponent(jLabel5)

            .addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGap(29, 29, 29)

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

            .addComponent(jButton1)

            .addComponent(jButton2)

            .addComponent(jButton3))

        .addGap(18, 18, 18)

        .addComponent(jButton4))

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 294, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addContainerGap(30, Short.MAX_VALUE))

    );

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

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) { // GEN-FIRST:event_jTable1MouseClicked

    DefaultTableModel model = (DefaultTableModel) jTable1.getModel();

    int selectedRowIndex = jTable1.getSelectedRow();

    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());

    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());

    jTextField3.setText(model.getValueAt(selectedRowIndex, 2).toString());

    jTextField4.setText(model.getValueAt(selectedRowIndex, 3).toString());

    jTextField5.setText(model.getValueAt(selectedRowIndex, 4).toString());

} // GEN-LAST:event_jTable1MouseClicked

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) { // GEN-FIRST:event_jButton1ActionPerformed

    String query = "INSERT INTO reservation VALUES ( " + jTextField1.getText() + ", " + jTextField2.getText() + ", " + jTextField3.getText() + ", " + jTextField4.getText() + ", " + jTextField5.getText() + " "
    );

    executeQuery(query, "Inserted");

} // GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) { // GEN-FIRST:event_jButton2ActionPerformed

```

```

        String query = "UPDATE reservation SET res_name= '"+jTextField3.getText()+"', res_lname= '"+jTextField4.getText()+"', res_isadult= '"+jTextField5.getText()+"' WHERE res_tr_id=
"+jTextField1.getText()+" and res_seatnum= '"+jTextField2.getText()+"'";

        executeQuery(query, "Updated");

    }//GEN-LAST:event_jButton2ActionPerformed

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_jButton3ActionPerformed

    String query = "DELETE FROM reservation WHERE res_tr_id = '"+jTextField1.getText()+" and res_seatnum= '"+jTextField2.getText()+"'";

    executeQuery(query, "Delete");

    }//GEN-LAST:event_jButton3ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_jButton4ActionPerformed

    this.setVisible(false);

    new Menu().setVisible(true);

    }//GEN-LAST:event_jButton4ActionPerformed

public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new Reservation().setVisible(true);

        }

    });

}

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;

```

```
private javax.swing.JTextField jTextField4;

private javax.swing.JTextField jTextField5;

// End of variables declaration//GEN-END:variables

}
```

```
package bd;
```

```
import java.sql.*;
import java.util.ArrayList;
import javax.swing.*;
import javax.swing.table.*;
```

```
public class Reservation_offers extends javax.swing.JFrame {
```

```
    public Reservation_offers() {
        initComponents();
        this.setLocationRelativeTo(null);
        DisplayTable();
    }
```

```
    public ArrayList<SetterReservation_offers> tableList(){
```

```
        ArrayList<SetterReservation_offers> tableList = new ArrayList<>();
        Connection con = (new GetConnection()).getConnection();
        String query = "select * from reservation_offers ";
        Statement st;
        ResultSet rs;
```

```
        try{
            st = con.createStatement();
            rs = st.executeQuery(query);
            SetterReservation_offers setter;
```



```

while(rs.next()){

    setter = new SetterReservation_offers(rs.getInt("res_off_id"),rs.getString("res_off_ln"), rs.getString("res_off_fn"), rs.getInt("res_off_off_id"), rs.getInt("res_off_prepay"));

    tableList.add(setter);

}

}catch(Exception ex){

    ex.printStackTrace();

}

return tableList;

}

```

```

private void DisplayTable(){

ArrayList<SetterReservation_offers> list = tableList();

DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

Object [] row = new Object [5];

for(int i=0; i<list.size(); i++){

    row[0]= list.get(i).getID();

    row[1]= list.get(i).getLn();

    row[2]= list.get(i).getFn();

    row[3]= list.get(i).getOff_id();

    row[4]= list.get(i).getPrepay();

    model.addRow(row);

}

}

```

```

public void executeQuery(String query, String message){

    Connection con = (new GetConnection()).getConnection();

    Statement st;

    try{

        st = con.createStatement();

        if((st.executeUpdate(query))==1){

            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

            model.setRowCount(0);

            DisplayTable();

            JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");

        }else{

            JOptionPane.showMessageDialog(null, "Data Not"+message);

        }

    }
}

```

```

    }

    }catch(Exception ex){

        ex.printStackTrace();

    }

}

@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();

    jLabel4 = new javax.swing.JLabel();

    jTextField1 = new javax.swing.JTextField();

    jTextField2 = new javax.swing.JTextField();

    jTextField3 = new javax.swing.JTextField();

    jTextField4 = new javax.swing.JTextField();

    jButton1 = new javax.swing.JButton();

    jButton2 = new javax.swing.JButton();

    jButton3 = new javax.swing.JButton();

    jScrollPane1 = new javax.swing.JScrollPane();

    jTable1 = new javax.swing.JTable();

    jLabel5 = new javax.swing.JLabel();

    jTextField5 = new javax.swing.JTextField();

    jButton4 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel1.setText("res_off_id");

    jLabel2.setText("res_off_ln");

    jLabel3.setText("res_off_fn");

    jLabel4.setText("res_off_off_id");

    jButton1.setText("INSERT");

```

```

jButton1.addActionListener(new java.awt.event.ActionListener() {

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

        jButton1ActionPerformed(evt);

    }

});

jButton2.setText("UPDATE");

jButton2.addActionListener(new java.awt.event.ActionListener() {

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

        jButton2ActionPerformed(evt);

    }

});

jButton3.setText("DELETE");

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

        jButton3ActionPerformed(evt);

    }

});

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "res_off_id", "res_off_ln", "res_off_fn", "res_off_off_id", "res_off_prepay"

    }

) {

    boolean[] canEdit = new boolean [] {

        false, false, false, false, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {

    public void mouseClicked(java.awt.event.MouseEvent evt) {

        jTable1MouseClicked(evt);

    }

});

```

```

    }

});

jScrollPane1.setViewportViewView(jTable1);

jLabel5.setText("res_off_prepay");

jButton4.setText("Menu");

jButton4.addActionListener(new java.awt.event.ActionListener() {

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

        jButton4ActionPerformed(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()

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

            .addGroup(layout.createSequentialGroup()

                .add(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                    .add(layout.createSequentialGroup()

                        .addGap(27, 27, 27)

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

                            .add(layout.createSequentialGroup()

                                .addComponent(jLabel4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                .addGap(4, 4, 4))

                            .addGroup(layout.createSequentialGroup()

                                .addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                .addGap(27, 27, 27))

                            .addGroup(layout.createSequentialGroup()

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

                                    .add(layout.createSequentialGroup()

                                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

                                            .addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED_SIZE, 63, javax.swing.GroupLayout.PREFERRED_SIZE)

                                            .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 65, javax.swing.GroupLayout.PREFERRED_SIZE))

                                        .addComponent(jLabel5))

                                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 49, Short.MAX_VALUE)))

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

                                    .add(layout.createSequentialGroup()

                                        .addComponent(jTextField1, javax.swing.GroupLayout.Alignment.TRAILING)

                                        .addComponent(jTextField2, javax.swing.GroupLayout.Alignment.TRAILING)

                                        .addComponent(jTextField3, javax.swing.GroupLayout.Alignment.TRAILING)


```

```

        .addComponent(jTextField4, javax.swing.GroupLayout.Alignment.TRAILING)

        .addComponent(jTextField5, javax.swing.GroupLayout.DEFAULT_SIZE, 95, Short.MAX_VALUE))

        .addGap(18, 18, 18))

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

        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

        .addComponent(jButton1)

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

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

            .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

            .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

        .addComponent(jButton3)

        .addGap(10, 10, 10)))

    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 707, 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())

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

                .addGroup(layout.createSequentialGroup())

                    .addGap(34, 34, 34)

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

                        .addComponent(jLabel1)

                        .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                        .addComponent(jLabel2)

                        .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                        .addComponent(jLabel3)

                        .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                        .addComponent(jLabel4)

                        .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

```

```

        .addComponent(jLabel5)

        .addComponent(jTextField5, 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(jButton1)

        .addComponent(jButton2)

        .addComponent(jButton3))

    .addGap(18, 18, 18)

    .addComponent(jButton4))

    .addGroup(layout.createSequentialGroup()

        .addGap(20, 20, 20)

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 456, javax.swing.GroupLayout.PREFERRED_SIZE)))

    .addContainerGap(30, Short.MAX_VALUE))

);

pack();

} // </editor-fold> // GEN-END: initComponents

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) // GEN-FIRST:event_jTable1MouseClicked

    DefaultTableModel model = (DefaultTableModel) jTable1.getModel();

    int selectedRowIndex = jTable1.getSelectedRow();

    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());

    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());

    jTextField3.setText(model.getValueAt(selectedRowIndex, 2).toString());

    jTextField4.setText(model.getValueAt(selectedRowIndex, 3).toString());

    jTextField5.setText(model.getValueAt(selectedRowIndex, 4).toString());

} // GEN-LAST:event_jTable1MouseClicked

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton1ActionPerformed

    String query = "INSERT INTO reservation_offers VALUES ( NULL, '"+jTextField2.getText()+"', '"+jTextField3.getText()+"', '"+jTextField4.getText()+"', '"+jTextField5.getText()+"' );";

    executeQuery(query, "Inserted");

} // GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton2ActionPerformed

    String query = "UPDATE reservation_offers SET res_off_ln= '"+jTextField2.getText()+"', res_off_fn= '"+jTextField3.getText()+"', res_off_off_id= '"+jTextField4.getText()+"', res_off_prepay= '"+jTextField5.getText()+"' WHERE res_off_id= '"+jTextField1.getText()+"' ";

    executeQuery(query, "Updated");

} // GEN-LAST:event_jButton2ActionPerformed

```

```

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jButton3ActionPerformed

    String query = "DELETE FROM reservation_offers WHERE res_off_id = "+jTextField1.getText()+" ";

    executeQuery(query, "Delete");

} //GEN-LAST:event_jButton3ActionPerformed


private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jButton4ActionPerformed

    this.setVisible(false);

    new Menu().setVisible(true);

} //GEN-LAST:event_jButton4ActionPerformed


public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new Reservation_offers().setVisible(true);

        }

    });

}


// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField4;
private javax.swing.JTextField jTextField5;
// End of variables declaration//GEN-END:variables

```

```
}
```

```
package bd;
```

```
import java.sql.Connection;
```

```
import java.sql.DriverManager;
```

```
import java.sql.ResultSet;
```

```
import java.sql.SQLException;
```

```
import java.sql.Statement;
```

```
import javax.swing.table.DefaultTableModel;
```

```
public class RevenueAndCost extends javax.swing.JFrame {
```

```
    public RevenueAndCost() {
```

```
        initComponents();
```

```
        this.setLocationRelativeTo(null);
```

```
        tableList();
```

```
    }
```

```
    private void tableList(){
```

```
        Connection con = (new GetConnection()).getConnection();
```

```
        String query = "select code1, sumofsalary, sumofcost*numofres as revenue\n" +
```

```
            "from\n" +
```

```
            "(select code1, sum(numof_tr_res) as numofres, sum(tr_cost) as sumofcost\n" +
```

```
            " from\n" +
```

```
            " (select br_code as code1, tr_id as id, count(*) as numof_tr_res, tr_cost\n" +
```

```
            " from trip inner join reservation on tr_id=res_tr_id\n" +
```

```
            "          inner join branch on br_code=tr_br_code\n" +
```

```
            " group by tr_id) as t1\n" +
```

```
            " group by code1) as t3\n" +
```



```

"inner join\n" +

"(select br_code as code2, sum(wrk_salary) as sumofsalary\n" +

"from branch right join worker on br_code=wrk_br_code\n" +

"group by br_code) as t2\n" +

"\n" +

"on code1=code2\n" +

"order by code1;";

```

```
Statement st;
```

```
ResultSet rs;
```

```
try{
```

```
    st = con.createStatement();
```

```
    rs = st.executeQuery(query);
```

```
    while(rs.next()){
```

```
        int data1 = rs.getInt("code1");
```

```
        float data2 = rs.getFloat("sumofsalary");
```

```
        float data3 = rs.getFloat("revenue");
```

```
        Object[] row1 = { data1, data2, data3};
```

```
        DefaultTableModel model1 = (DefaultTableModel) jTable1.getModel();
```

```
        model1.addRow(row1);
```

```
    }
```

```
}catch(Exception ex){
```

```
    ex.printStackTrace();
```

```
}
```

```
}
```

```
@SuppressWarnings("unchecked")
```

```
// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN: initComponents
```

```
private void initComponents() {
```

```
    jButton1 = new javax.swing.JButton();
```

```

jScrollPane1 = new javax.swing.JScrollPane();

jTable1 = new javax.swing.JTable();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

jButton1.setText("Menu");

jButton1.addActionListener(new java.awt.event.ActionListener() {

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

        jButton1ActionPerformed(evt);

    }

});

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "br_code", "salaries", "revenue"

    }

) {

    boolean[] canEdit = new boolean [] {

        false, false, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

jScrollPane1.setViewportView(jTable1);

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)

                .add(layout.createSequentialGroup()

                    .add(layout.createSequentialGroup()

                        .add(layout.createSequentialGroup()

                            .addGap(29, 29, 29)

                            .addComponent(jButton1)


```

```

        .addGroup(layout.createSequentialGroup())

        .addGap(78, 78, 78)

        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addContainerGap(83, Short.MAX_VALUE))

);

layout.setVerticalGroup(

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

        .addGroup(layout.createSequentialGroup())

            .addGap(24, 24, 24)

            .addComponent(jButton1)

            .addGap(32, 32, 32)

            .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 369, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addContainerGap(54, Short.MAX_VALUE))

);

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

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton1ActionPerformed
{
    this.setVisible(false);

    new Menu().setVisible(true);
} // GEN-LAST:event_jButton1ActionPerformed

public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new RevenueAndCost().setVisible(true);

        }

    });
}

// Variables declaration - do not modify // GEN-BEGIN:variables
private javax.swing.JButton jButton1;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTable jTable1;

// End of variables declaration // GEN-END:variables
}

```

```
package bd;
```

```
import java.sql.Connection;
```

```
import java.sql.DriverManager;
```

```
import java.sql.ResultSet;
```

```
import java.sql.SQLException;
```

```
import java.sql.Statement;
```

```
import java.util.ArrayList;
```

```
import javax.swing.JOptionPane;
```

```
import javax.swing.table.DefaultTableModel;
```

```
public class SearchTrip extends javax.swing.JFrame {
```

```
    public SearchTrip() {
```

```
        initComponents();
```

```
        this.setLocationRelativeTo(null);
```

```
    }
```

```
    private void tableList(){
```

```
        Connection con = (new GetConnection()).getConnection();
```

```
        String query = "call SearchTrip('"+jTextField1.getText()+"', '"+jTextField2.getText()+"', '"+jTextField3.getText()+"')";
```

```
        Statement st;
```

```
        ResultSet rs;
```

```
        try{
```

```
            st = con.createStatement();
```

```
            rs = st.executeQuery(query);
```

```

int count=0;

if(rs.next()){

    while(rs.next()){

        int data1 = rs.getInt("tr_id1");

        float data2 = rs.getFloat("tr_cost");

        int data3 = rs.getInt("tr_maxseats");

        String data4 = rs.getString("wrk_name");

        String data5 = rs.getString("wrk_lname");

        String data6 = rs.getString("tr_departure");

        String data7 = rs.getString("tr_return");

        int data8 = rs.getInt("tr_id2");

        int data9 = rs.getInt("numofreservations");

        int data10 = rs.getInt("availableseats");

        Object[] row1 = { data1, data2, data3, data4, data5, data6, data7};

        Object[] row2 = { data8, data9, data10};

        DefaultTableModel model1 = (DefaultTableModel) jTable1.getModel();

        DefaultTableModel model2 = (DefaultTableModel) jTable2.getModel();

        model1.addRow(row1);

        if(count%2==0)

            model2.addRow(row2);

        count++;

    }

}else{

    JOptionPane.showMessageDialog(this, "There is no trip");

}

}catch(Exception ex){

    ex.printStackTrace();

}

}

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-BEGIN: initComponents

private void initComponents() {

```

```

jScrollPane1 = new javax.swing.JScrollPane();

jTable1 = new javax.swing.JTable();

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

jLabel3 = new javax.swing.JLabel();

jTextField1 = new javax.swing.JTextField();

jTextField2 = new javax.swing.JTextField();

jTextField3 = new javax.swing.JTextField();

jButton1 = new javax.swing.JButton();

jScrollPane2 = new javax.swing.JScrollPane();

jTable2 = new javax.swing.JTable();

jButton2 = new javax.swing.JButton();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);


jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {

        },
    new String [] {
        "trip id", "trip cost", "max seats", "driver/guide name", "driver/guide ln", "departure", "return"
    }
) {
    boolean[] canEdit = new boolean [] {
        false, false, false, false, false, false, false
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return canEdit [columnIndex];
    }
});

jScrollPane1.setViewportView(jTable1);


jLabel1.setText("branch id");


jLabel2.setText("start date");


jLabel3.setText("end date");

```

```

jTextField1.setText("1");

jTextField2.setText("2022-11-10 09:00:00");

jTextField2.addActionListener(new java.awt.event.ActionListener() {

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

        jTextField2ActionPerformed(evt);

    }

});

jTextField3.setText("2022-12-13 12:00:00");


jButton1.setText("SELECT");

jButton1.addActionListener(new java.awt.event.ActionListener() {

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

        jButton1ActionPerformed(evt);

    }

});


jTable2.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "trp id", "num of reservations", "avialable seats"

    }

) {

    boolean[] canEdit = new boolean [] {

        false, true, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

jScrollPane2.setViewportViewView(jTable2);


jButton2.setText("Menu");

jButton2.addActionListener(new java.awt.event.ActionListener() {

```

```

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

    jButton2ActionPerformed(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()

            .addGap(18, 18, 18)

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

                .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, 61, Short.MAX_VALUE)

                .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                .addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

            .addGap(18, 18, 18)

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

                .addComponent(jTextField2)

                .addComponent(jTextField3)

                .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, 152, javax.swing.GroupLayout.PREFERRED_SIZE)

                .addGroup(layout.createSequentialGroup()

                    .addGap(32, 32, 32)

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

                        .addComponent(jButton1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                        .addComponent(jButton2, javax.swing.GroupLayout.PREFERRED_SIZE, 83, javax.swing.GroupLayout.PREFERRED_SIZE))))

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 48, Short.MAX_VALUE)

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

                    .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, 487, javax.swing.GroupLayout.PREFERRED_SIZE)

                    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 921, javax.swing.GroupLayout.PREFERRED_SIZE))

                .addGap(33, 33, 33))

        )

);

layout.setVerticalGroup(

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

        .addGroup(layout.createSequentialGroup()

            .addGap(54, 54, 54)

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

                .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 190, javax.swing.GroupLayout.PREFERRED_SIZE)

                .addGroup(layout.createSequentialGroup()

                    .addGap(18, 18, 18)

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

```



```

        .addComponent(jTextField1)

        .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 22, javax.swing.GroupLayout.PREFERRED_SIZE))

    .addGap(18, 18, 18)

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

        .addComponent(jLabel2)

        .addComponent(jTextField2, 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(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))

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

        .addGroup(layout.createSequentialGroup()

            .addGap(57, 57, 57)

            .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, 167, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGroup(layout.createSequentialGroup()

            .addGap(36, 36, 36)

            .addComponent(jButton1)

            .addGap(18, 18, 18)

            .addComponent(jButton2)))

    .addContainerGap(61, Short.MAX_VALUE))

);

pack();

}

// </editor-fold> //GEN-END: initComponents

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton1ActionPerformed
    tableList();
} //GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton2ActionPerformed
    this.setVisible(false);
    new Menu().setVisible(true);
} //GEN-LAST:event_jButton2ActionPerformed

private void jTextField2ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jTextField2ActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_jTextField2ActionPerformed

```

```

public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new SearchTrip().setVisible(true);

        }

    });

}

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JScrollPane jScrollPane2;

private javax.swing.JTable jTable1;

private javax.swing.JTable jTable2;

private javax.swing.JTextField jTextField1;

private javax.swing.JTextField jTextField2;

private javax.swing.JTextField jTextField3;

// End of variables declaration//GEN-END:variables
}

```

```

package bd;

```

```

public class SetterAdmin {

    private String adm_AT;

    private String adm_type;

    private String adm_diploma;

    public SetterAdmin(String AT, String type, String diploma){

```

```

        this.adm_AT = AT;

        this.adm_type = type;

        this.adm_diploma = diploma;
    }

    public String getAT(){

        return adm_AT;
    }

    public String getType(){

        return adm_type;
    }

    public String getDiploma(){

        return adm_diploma;
    }
}

```

```

package bd;

```

```

public class SetterBranch {

    private int br_code;

    private String br_street;

    private int br_num;

    private String br_city;

    public SetterBranch(int code, String street, int num, String city){

        this.br_code = code;

        this.br_street = street;

        this.br_num = num;

        this.br_city = city;
    }
}

```

```
public int getCode(){  
    return br_code;  
}
```

```
public String getStreet(){  
    return br_street;  
}
```

```
public int getNum(){  
    return br_num;  
}
```

```
public String getCity(){  
    return br_city;  
}  
}
```

```
package bd;
```

```
public class SetterDestination {  
    private int dst_id;  
    private String dst_name;  
    private String dst_descr;  
    private String dst_rtype;  
    private String dst_language;  
    private int dst_location;  
  
    public SetterDestination(int id, String name, String descr, String rtype, String language, int location){  
        this.dst_id = id;  
        this.dst_name = name;  
        this.dst_descr = descr;  
        this.dst_rtype = rtype;  
        this.dst_language = language;  
    }  
}
```

```

        this.dst_location = location;
    }

    public int getID(){
        return dst_id;
    }

    public String getName(){
        return dst_name;
    }

    public String getDescr(){
        return dst_descr;
    }

    public String getRtype(){
        return dst_rtype;
    }

    public String getLanguage(){
        return dst_language;
    }

    public int getLocation(){
        return dst_location;
    }
}

```

```

package bd;

```

```

public class SetterDriver {
    private String drv_AT;
    private String drv_license;

```

```

private String drv_route;

private int drv_experience;

public SetterDriver(String AT, String license, String route, int experience){

    this.drv_AT = AT;

        this.drv_license = license;

        this.drv_route = route;

        this.drv_experience = experience;

}

public String getAT(){

    return drv_AT;

}

public String getLicense(){

    return drv_license;

}

public String getRoute(){

    return drv_route;

}

public int getExperience(){

    return drv_experience;

}

}

```

```

package bd;

```

```

public class SetterEvent {

    private int ev_tr_id;

    private String ev_start;

    private String ev_end;

```

```

private String ev_descr;

public SetterEvent(int id, String start, String end, String descr){

    this.ev_tr_id = id;

    this.ev_start = start;

    this.ev_end = end;

    this.ev_descr = descr;

}

public int getID(){

    return ev_tr_id;

}

public String getStart(){

    return ev_start;

}

public String getEnd(){

    return ev_end;

}

public String getDescr(){

    return ev_descr;

}

}

```

```

package bd;

```

```

public class SetterGuide {

    private String gui_AT;

    private String gui_cv;

    public SetterGuide( String AT, String cv){

```

```
    this.gui_AT = AT;

    this.gui_cv = cv;
}
```

```
public String getAT(){

    return gui_AT;
}
```

```
public String getCV(){

    return gui_cv;
}
```

```
}
```

```
package bd;
```

```
public class SetterIT {

    private String IT_AT;

    private String password;

    private String start_date;

    private String end_date;
```

```
public SetterIT(String AT, String pass, String start,String end){

    this.IT_AT = AT;

    this.password = pass;

    this.start_date = start;

    this.end_date = end;
}
```

```
public String getAT(){

    return IT_AT;
}
```



```
public String getPassword(){  
    return password;  
}
```

```
public String getStart(){  
    return start_date;  
}
```

```
public String getEnd(){  
    return end_date;  
}  
}
```

```
package bd;
```

```
public class SetterLanguage {  
    private String lng_gui_AT;  
    private String lng_language;  
  
    public SetterLanguage( String AT, String language){  
        this.lng_gui_AT = AT;  
        this.lng_language = language;  
    }
```

```
public String getAT(){  
    return lng_gui_AT;  
}
```

```
public String getLanguage(){  
    return lng_language;  
}
```

```
}
```

```
package bd;
```

```
public class SetterManages {  
    private String mng_adm_AT;  
    private int mng_br_code;
```

```
    public SetterManages(String AT, int code){  
        this.mng_adm_AT = AT;  
        this.mng_br_code = code;  
    }
```

```
    public String getAT(){  
        return mng_adm_AT;  
    }
```

```
    public int getCode(){  
        return mng_br_code;  
    }
```

```
}
```

```
package bd;
```

```
public class SetterOffers {  
    private int off_id;  
    private String off_start;
```

```
private String off_end;

private int off_cost;

private int off_dst_id;


public SetterOffers(int id, String start, String end, int cost, int dst_id){

    this.off_id = id;

    this.off_start = start;

    this.off_end = end;

    this.off_cost = cost;

    this.off_dst_id = dst_id;

}


public int getID(){

    return off_id;

}


public String getStart(){

    return off_start;

}


public String getEnd(){

    return off_end;

}


public int getCost(){

    return off_cost;

}


public int getDst_id(){

    return off_dst_id;

}

}
```

```
package bd;
```

```

public class SetterPhones {

    private int ph_br_code;

    private String ph_number;


    public SetterPhones( int code,String number){

        this.ph_br_code = code;

        this.ph_number = number;

    }


    public int getCode(){

        return ph_br_code;

    }


    public String getNumber(){

        return ph_number;

    }

}

```

```

package bd;

```

```

public class SetterReservation {

    private int res_tr_id;

    private int res_seatnum;

    private String res_name;

    private String res_lname;

    private String res_isadult;


    public SetterReservation(int id, int seatnum, String name, String lname, String isadult){

        this.res_tr_id = id;

        this.res_seatnum = seatnum;

    }

}

```

```

        this.res_name = name;

        this.res_lname = lname;

        this.res_isadult = isadult;
    }

    public int getID(){
        return res_tr_id;
    }

    public int getSeatnum(){
        return res_seatnum;
    }

    public String getName(){
        return res_name;
    }

    public String getLname(){
        return res_lname;
    }

    public String getIsadult(){
        return res_isadult;
    }
}

```

```

package bd;

```

```

public class SetterReservation_offers {

    private int res_off_id;

    private String res_off_lname;

    private String res_off_fn;

    private int res_off_off_id;

```

```

private int res_off_prepay;

public SetterReservation_offers(int id, String lname, String fname, int off_id, int prepay){

    this.res_off_id = id;

    this.res_off_ln = lname;

    this.res_off_fn = fname;

    this.res_off_off_id = off_id;

    this.res_off_prepay = prepay;
}

public int getID(){

    return res_off_id;

}

public String getLn(){

    return res_off_ln;

}

public String getFn(){

    return res_off_fn;

}

public int getOff_id(){

    return res_off_id;

}

public int getPrepay(){

    return res_off_prepay;

}
}

```

```
package bd;
```

```

public class SetterTravel_to {

    private int to_dst_id;

    private int to_tr_id;

    private String to_arrival;

    private String to_departure;


    public SetterTravel_to(int id_dst, int id_tr, String arrival, String departure){

        this.to_dst_id = id_dst;

        this.to_tr_id = id_tr;

        this.to_arrival = arrival;

        this.to_departure = departure;

    }


    public int getID_dst(){

        return to_dst_id;

    }


    public int getID_tr(){

        return to_tr_id;

    }


    public String getArrival(){

        return to_arrival;

    }


    public String getDeparture(){

        return to_departure;

    }

}

```

```

package bd;

```

```

public class SetterTrip {

```

```

private int tr_id;

private String tr_departure;

private String tr_return;

private int tr_maxseats;

private float tr_cost;

private int tr_br_code;

private String tr_gui_AT;

private String tr_drv_AT;


public SetterTrip(int id, String departure, String return_date, int maxseats, float cost, int code, String gui_AT, String drv_AT){

    this.tr_id = id;

    this.tr_departure = departure;

    this.tr_return = return_date;

    this.tr_maxseats = maxseats;

    this.tr_cost = cost;

    this.tr_br_code = code;

    this.tr_gui_AT = gui_AT;

    this.tr_drv_AT = drv_AT;

}


public int getID(){

    return tr_id;

}

public String getDeparture(){

    return tr_departure;

}


public String getReturn_date(){

    return tr_return;

}


public int getMaxseats(){

    return tr_maxseats;

}


public float getCost(){

    return tr_cost;

}

```



```

public int getCode(){
    return tr_br_code;
}

public String getGui_AT(){
    return tr_gui_AT;
}

public String getDrv_AT(){
    return tr_drv_AT;
}
}

```

```

package bd;

```

```

public class SetterWorker {
    private String wrk_AT;
    private String wrk_name;
    private String wrk_lname;
    private float wrk_salary;
    private int wrk_br_code;

    public SetterWorker(String AT, String name, String lname, float salary, int code){
        this.wrk_AT = AT;
        this.wrk_name = name;
        this.wrk_lname = lname;
        this.wrk_salary = salary;
        this.wrk_br_code = code;
    }

    public String getAT(){

```

```

        return wrk_AT;
    }

    public String getName(){
        return wrk_name;
    }

    public String getLastName(){
        return wrk_lname;
    }

    public float getSalary(){
        return wrk_salary;
    }

    public int getCode(){
        return wrk_br_code;
    }
}

```

```

package bd;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.table.DefaultTableModel;

public class ShowLog extends javax.swing.JFrame {

    public ShowLog() {

```

```

initComponents();

this.setLocationRelativeTo(null);

tableList();
}

private void tableList(){

    Connection con = (new GetConnection()).getConnection();

    String query = "select action_taken, who, chosentable, cur_datetime from log; " ;

    Statement st;
    ResultSet rs;

    try{
        st = con.createStatement();
        rs = st.executeQuery(query);

        while(rs.next()){
            String data1 = rs.getString("action_taken");
            String data2 = rs.getString("who");
            String data3 = rs.getString("chosentable");
            String data4 = rs.getString("cur_datetime");

            Object[] row = { data1, data2, data3, data4};

            DefaultTableModel model = (DefaultTableModel) jTable1.getModel();

            model.addRow(row);
        }

    }catch(Exception ex){
        ex.printStackTrace();
    }
}

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">

```

```

private void initComponents() {

    jScrollPane1 = new javax.swing.JScrollPane();

    jTable1 = new javax.swing.JTable();

    jButton1 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jTable1.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {

        },
        new String [] {
            "action", "lname", "table", "datetime"
        }
    ) {
        boolean[] canEdit = new boolean [] {
            false, false, true, false
        };

        public boolean isCellEditable(int rowIndex, int columnIndex) {
            return canEdit [columnIndex];
        }
    });

    jScrollPane1.setViewportView(jTable1);

    jButton1.setText("Menu");

    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()
                .addContainerGap()
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(jButton1)
                )
            )
    );
}

```

```

        .addGroup(layout.createSequentialGroup())

        .addGap(23, 23, 23)

        .addComponent(jButton1))

    .addGroup(layout.createSequentialGroup())

    .addGap(58, 58, 58)

    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 756, javax.swing.GroupLayout.PREFERRED_SIZE)))

    .addContainerGap(64, Short.MAX_VALUE))

);

layout.setVerticalGroup(

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

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

    .addGap(26, 26, 26)

    .addComponent(jButton1)

    .addGap(28, 28, 28)

    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 329, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addContainerGap(36, Short.MAX_VALUE))

);

pack();

} // </editor-fold> // GEN-END: initComponents

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton1ActionPerformed

    this.setVisible(false);

    new Menu().setVisible(true);

} // GEN-LAST:event_jButton1ActionPerformed

```

```

public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new ShowLog().setVisible(true);

        }

    });

}

```

```

// Variables declaration - do not modify // GEN-BEGIN:variables

private javax.swing.JButton jButton1;

private javax.swing.JScrollPane jScrollPane1;

```

```
private javax.swing.JTable jTable1;

// End of variables declaration//GEN-END:variables

}
```

```
package bd;
```

```
import java.sql.*;
import java.util.ArrayList;
import javax.swing.*;
import javax.swing.table.*;
```

```
public class Travel_to extends javax.swing.JFrame {
```

```
    public Travel_to() {
        initComponents();
        this.setLocationRelativeTo(null);
        DisplayTable();
    }
```

```
    public ArrayList<SetterTravel_to> tableList(){
```

```
        ArrayList<SetterTravel_to> tableList = new ArrayList<>();
        Connection con = (new GetConnection()).getConnection();
        String query = "select * from travel_to ";
        Statement st;
        ResultSet rs;
```

```
        try{
            st = con.createStatement();
            rs = st.executeQuery(query);
            SetterTravel_to setter;
```

```

while(rs.next()){

    setter = new SetterTravel_to(rs.getInt("_dst_id"), rs.getInt("_tr_id"),rs.getString("_arrival"),rs.getString("_departure"));

    tableList.add(setter);

}

}catch(Exception ex){

    ex.printStackTrace();

}

return tableList;

}

```

```

private void DisplayTable(){

ArrayList<SetterTravel_to> list = tableList();

DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

Object [] row = new Object [4];

for(int i=0; i<list.size(); i++){

    row[0]= list.get(i).getID_dst();

    row[1]= list.get(i).getID_tr();

    row[2]= list.get(i).getArrival();

    row[3]= list.get(i).getDeparture();

    model.addRow(row);

}

}

```

```

public void executeQuery(String query, String message){

    Connection con = (new GetConnection()).getConnection();

    Statement st;

    try{

        st = con.createStatement();

        if((st.executeUpdate(query))==1){

            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

            model.setRowCount(0);

            DisplayTable();

            JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");

        }else{

            JOptionPane.showMessageDialog(null, "Data Not"+message);

        }

    }

```

```

    }catch(Exception ex){

        ex.printStackTrace();

    }

}

@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();

    jLabel4 = new javax.swing.JLabel();

    jTextField1 = new javax.swing.JTextField();

    jTextField2 = new javax.swing.JTextField();

    jTextField3 = new javax.swing.JTextField();

    jTextField4 = new javax.swing.JTextField();

    jButton1 = new javax.swing.JButton();

    jButton2 = new javax.swing.JButton();

    jButton3 = new javax.swing.JButton();

    jScrollPane1 = new javax.swing.JScrollPane();

    jTable1 = new javax.swing.JTable();

    jButton4 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel1.setText("to_tr_id");

    jLabel2.setText("to_dst_id");

    jLabel3.setText("to_arrival");

    jLabel4.setText("to_departure");

    jButton1.setText("INSERT");

    jButton1.addActionListener(new java.awt.event.ActionListener() {

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

            jButton1ActionPerformed(evt);

```



```

    }

});

jButton2.setText("UPDATE");

jButton2.addActionListener(new java.awt.event.ActionListener() {

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

        jButton2ActionPerformed(evt);

    }

});

jButton3.setText("DELETE");

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

        jButton3ActionPerformed(evt);

    }

});

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "to_dst_id", "to_tr_id", "to_arrival", "to_departure"

    }

) {

    boolean[] canEdit = new boolean [] {

        false, false, true, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {

    public void mouseClicked(java.awt.event.MouseEvent evt) {

        jTable1MouseClicked(evt);

    }

});

jScrollPane1.setViewportView(jTable1);

```

```

jButton4.setText("Menu");

jButton4.addActionListener(new java.awt.event.ActionListener() {

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

        jButton4ActionPerformed(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()

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

                .addGroup(layout.createSequentialGroup()

                    .add(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                        .add(layout.createSequentialGroup()

                            .addGap(26, 26, 26)

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

                                .add(layout.createSequentialGroup()

                                    .addComponent(jLabel4, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT_SIZE, 87, Short.MAX_VALUE)

                                    .addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED_SIZE, 76, javax.swing.GroupLayout.PREFERRED_SIZE)

                                    .addComponent(jLabel2, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

                                .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 87, javax.swing.GroupLayout.PREFERRED_SIZE))

                            .addGap(20, 20, 20)

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

                                .addComponent(jTextField3, javax.swing.GroupLayout.DEFAULT_SIZE, 130, Short.MAX_VALUE)

                                .addComponent(jTextField2)

                                .addComponent(jTextField4)

                                .addComponent(jTextField1, javax.swing.GroupLayout.Alignment.TRAILING)))

                            .addGroup(layout.createSequentialGroup()

                                .addContainerGap()

                                .addComponent(jButton1)

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

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

                                    .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                    .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

                                .addComponent(jButton3)))

                            .addGap(18, 18, 18)

                            .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 565, javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addContainerGap())

);

layout.setVerticalGroup(

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

        .addGroup(layout.createSequentialGroup())

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

                .addGroup(layout.createSequentialGroup())

                    .addGap(26, 26, 26)

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

                        .addComponent(jLabel1)

                        .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                        .addComponent(jLabel2)

                        .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                        .addComponent(jLabel3)

                        .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                        .addComponent(jLabel4)

                        .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addGap(44, 44, 44)

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

                        .addComponent(jButton1)

                        .addComponent(jButton2)

                        .addComponent(jButton3))

                    .addGap(18, 18, 18)

                    .addComponent(jButton4))

            .addGroup(layout.createSequentialGroup())

                .addContainerGap()

                .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 275, javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addContainerGap(19, Short.MAX_VALUE))

);

pack();

} // </editor-fold> // GEN-END: initComponents

```

```
private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {GEN-FIRST:event_jTable1MouseClicked
```

```
    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();
```

```
    int selectedRowIndex = jTable1.getSelectedRow();
```

```
    jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());
```

```
    jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());
```

```
    jTextField3.setText(model.getValueAt(selectedRowIndex, 2).toString());
```

```
    jTextField4.setText(model.getValueAt(selectedRowIndex, 3).toString());
```

```
}GEN-LAST:event_jTable1MouseClicked
```

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jButton1ActionPerformed
```

```
    String query = "INSERT INTO travel_to VALUES ( "+jTextField1.getText()+" , "+jTextField2.getText()+" , "+jTextField3.getText()+" , "+jTextField4.getText()+" )";
```

```
    executeQuery(query, "Inserted");
```

```
}GEN-LAST:event_jButton1ActionPerformed
```

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jButton2ActionPerformed
```

```
    String query = "UPDATE travel_to SET to_dst_id= "+jTextField2.getText()+" , to_arrival= "+jTextField3.getText()+" , to_departure= "+jTextField4.getText()+" WHERE to_tr_id= "+jTextField1.getText()+" ";
```

```
    executeQuery(query, "Updated");
```

```
}GEN-LAST:event_jButton2ActionPerformed
```

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jButton3ActionPerformed
```

```
    String query = "DELETE FROM travel_to WHERE to_tr_id= "+jTextField1.getText()+" ";
```

```
    executeQuery(query, "Delete");
```

```
}GEN-LAST:event_jButton3ActionPerformed
```

```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_jButton4ActionPerformed
```

```
    this.setVisible(false);
```

```
    new Menu().setVisible(true);
```

```
}GEN-LAST:event_jButton4ActionPerformed
```

```
public static void main(String args[]) {
```

```
    java.awt.EventQueue.invokeLater(new Runnable() {
```

```
        public void run() {
```

```
            new Travel_to().setVisible(true);
```

```
        }
```

```
    });
```

```

}

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JButton jButton3;

private javax.swing.JButton jButton4;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTable jTable1;

private javax.swing.JTextField jTextField1;

private javax.swing.JTextField jTextField2;

private javax.swing.JTextField jTextField3;

private javax.swing.JTextField jTextField4;

// End of variables declaration//GEN-END:variables
}

```

```

package bd;

```

```

import java.sql.*;

import java.util.ArrayList;

import javax.swing.*;

import javax.swing.table.*;

```

```

public class Trip extends javax.swing.JFrame {

```

```

    public Trip() {

        initComponents();

        this.setLocationRelativeTo(null);

        DisplayTable();
    }

```

```
}
```

```
public ArrayList<SetterTrip> tableList(){
```

```
    ArrayList<SetterTrip> tableList = new ArrayList<>();
```

```
    Connection con = (new GetConnection()).getConnection();
```

```
    String query = "select * from trip ";
```

```
    Statement st;
```

```
    ResultSet rs;
```

```
    try{
```

```
        st = con.createStatement();
```

```
        rs = st.executeQuery(query);
```

```
        SetterTrip setter;
```

```
        while(rs.next()){
```

```
            setter = new
```

```
SetterTrip(rs.getInt("tr_id"),rs.getString("tr_departure"),rs.getString("tr_return"),rs.getInt("tr_maxseats"),rs.getInt("tr_cost"),rs.getInt("tr_br_code"),rs.getString("tr_gui_AT"),rs.getString("tr_drv_AT"));
```

```
            tableList.add(setter);
```

```
        }
```

```
    }catch(Exception ex){
```

```
        ex.printStackTrace();
```

```
    }
```

```
    return tableList;
```

```
}
```

```
private void DisplayTable(){
```

```
    ArrayList<SetterTrip> list = tableList();
```

```
    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();
```

```
    Object [] row = new Object [8];
```

```
    for(int i=0; i<list.size(); i++){
```

```
        row[0]= list.get(i).getID();
```

```
        row[1]= list.get(i).getDeparture();
```

```
        row[2]= list.get(i).getReturn_date();
```

```
        row[3]= list.get(i).getMaxseats();
```

```
        row[4]= list.get(i).getCost();
```

```
        row[5]= list.get(i).getCode();
```

```
        row[6]= list.get(i).getGui_AT();
```

```
        row[7]= list.get(i).getDrv_AT();
```

```

        model.addRow(row);
    }
}

public void executeQuery(String query, String message){
    Connection con = (new GetConnection()).getConnection();
    Statement st;

    try{
        st = con.createStatement();
        if((st.executeUpdate(query))==1){
            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();
            model.setRowCount(0);
            DisplayTable();
            JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");
        }else{
            JOptionPane.showMessageDialog(null, "Data Not"+message);
        }
    }catch(Exception ex){
        ex.printStackTrace();
        JOptionPane.showMessageDialog(null,ex.getMessage() );
    }
}

```

```

@SuppressWarnings("unchecked")

```

```

// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents

```

```

private void initComponents() {

```

```

    jLabel9 = new javax.swing.JLabel();
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    jLabel6 = new javax.swing.JLabel();

```

```

jLabel7 = new javax.swing.JLabel();

jLabel8 = new javax.swing.JLabel();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

jButton3 = new javax.swing.JButton();

jTextField1 = new javax.swing.JTextField();

jTextField2 = new javax.swing.JTextField();

jTextField3 = new javax.swing.JTextField();

jTextField4 = new javax.swing.JTextField();

jTextField5 = new javax.swing.JTextField();

jTextField6 = new javax.swing.JTextField();

jTextField7 = new javax.swing.JTextField();

jTextField8 = new javax.swing.JTextField();

jScrollPane1 = new javax.swing.JScrollPane();

jTable1 = new javax.swing.JTable();

jButton4 = new javax.swing.JButton();


jLabel9.setText("jLabel9");


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);


jLabel1.setText("tr_id");


jLabel2.setText("tr_departure");


jLabel3.setText("tr_return");


jLabel4.setText("tr_maxseats");


jLabel5.setText("tr_cost");


jLabel6.setText("tr_br_code");


jLabel7.setText("tr_gui_AT");


jLabel8.setText("tr_drv_AT");


jButton1.setText("INSERT");

jButton1.addActionListener(new java.awt.event.ActionListener() {

```



```

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

        jButton1ActionPerformed(evt);

    }

});

jButton2.setText("UPDATE");

jButton2.addActionListener(new java.awt.event.ActionListener() {

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

        jButton2ActionPerformed(evt);

    }

});

jButton3.setText("DELETE");

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

        jButton3ActionPerformed(evt);

    }

});

jTextField5.addActionListener(new java.awt.event.ActionListener() {

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

        jTextField5ActionPerformed(evt);

    }

});

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "tr_id", "tr_departure", "tr_return", "tr_maxseats", "tr_cost", "tr_br_code", "tr_gui_AT", "tr_drv_AT"

    }

) {

    boolean[] canEdit = new boolean [] {

        false, false, false, false, false, false, false, true

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

```



```

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

            .addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT_SIZE, 120, Short.MAX_VALUE)

            .addComponent(jTextField2)

            .addComponent(jTextField8))

        .addComponent(jTextField3, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.PREFERRED_SIZE, 120,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addComponent(jTextField4, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.PREFERRED_SIZE, 120,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addComponent(jTextField5, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.PREFERRED_SIZE, 120, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addComponent(jTextField7, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.PREFERRED_SIZE, 120, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addComponent(jTextField6, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.PREFERRED_SIZE, 120, javax.swing.GroupLayout.PREFERRED_SIZE)))

    .addGroup(layout.createSequentialGroup())

        .addContainerGap()

        .addComponent(jButton1)

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

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

            .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

            .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

        .addComponent(jButton3)))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 43, Short.MAX_VALUE)

    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 978, javax.swing.GroupLayout.PREFERRED_SIZE)

    .addGap(22, 22, 22))

);

layout.setVerticalGroup(

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

        .addGroup(layout.createSequentialGroup())

            .addGap(19, 19, 19)

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

                .addComponent(jLabel1)

                .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                .addComponent(jLabel2)

                .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

                .addComponent(jLabel3)

                .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

```

```

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

    .addComponent(jLabel4)

    .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

    .addComponent(jLabel5)

    .addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

.addGap(6, 6, 6)

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

    .addComponent(jLabel6)

    .addComponent(jTextField6, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

    .addComponent(jLabel7)

    .addComponent(jTextField7, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

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

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

    .addComponent(jLabel8)

    .addComponent(jTextField8, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

.addGap(68, 68, 68)

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

    .addComponent(jButton1)

    .addComponent(jButton2)

    .addComponent(jButton3))

.addGap(18, 18, 18)

.addComponent(jButton4)

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

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

.addContainerGap(19, Short.MAX_VALUE)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 479, javax.swing.GroupLayout.PREFERRED_SIZE)

.addContainerGap(23, Short.MAX_VALUE))

);

pack();

} // </editor-fold> // GEN-END: initComponents

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) // GEN-FIRST:event_jTable1MouseClicked

    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();

```

```

int selectedIndex = jTable1.getSelectedRow();

jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());
jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());
jTextField3.setText(model.getValueAt(selectedRowIndex, 2).toString());
jTextField4.setText(model.getValueAt(selectedRowIndex, 3).toString());
jTextField5.setText(model.getValueAt(selectedRowIndex, 4).toString());
jTextField6.setText(model.getValueAt(selectedRowIndex, 5).toString());
jTextField7.setText(model.getValueAt(selectedRowIndex, 6).toString());
jTextField8.setText(model.getValueAt(selectedRowIndex, 7).toString());

} //GEN-LAST:event_jTable1MouseClicked

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton1ActionPerformed
    String query = "INSERT INTO trip VALUES ( NULL, '"+jTextField2.getText()+"', '"+jTextField3.getText()+"', '"+jTextField4.getText()+"', '"+jTextField5.getText()+"', '"+jTextField6.getText()+"', '"+jTextField7.getText()+"', '"+jTextField8.getText()+"'";

    executeQuery(query, "Inserted");

} //GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton2ActionPerformed
    String query = "UPDATE trip SET tr_departure= '"+jTextField2.getText()+"', tr_return= '"+jTextField3.getText()+"', tr_maxseats= '"+jTextField4.getText()+"', tr_cost= '"+jTextField5.getText()+"', tr_br_code= '"+jTextField6.getText()+"', tr_gui_AT= '"+jTextField7.getText()+"', tr_drv_AT= '"+jTextField8.getText()+"' WHERE tr_id= '"+jTextField1.getText()+"' ";

    executeQuery(query, "Updated");

} //GEN-LAST:event_jButton2ActionPerformed

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton3ActionPerformed
    String query = "DELETE FROM trip WHERE tr_id= '"+jTextField1.getText()+"' ";

    executeQuery(query, "Delete");

} //GEN-LAST:event_jButton3ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jButton4ActionPerformed
    this.setVisible(false);

    new Menu().setVisible(true); } //GEN-LAST:event_jButton4ActionPerformed

private void jTextField5ActionPerformed(java.awt.event.ActionEvent evt) //GEN-FIRST:event_jTextField5ActionPerformed
    // TODO add your handling code here:

} //GEN-LAST:event_jTextField5ActionPerformed

public static void main(String args[]) {

```

```

        java.awt.EventQueue.invokeLater(new Runnable() {

            public void run() {

                new Trip().setVisible(true);

            }

        });
    }

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel9;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField4;
private javax.swing.JTextField jTextField5;
private javax.swing.JTextField jTextField6;
private javax.swing.JTextField jTextField7;
private javax.swing.JTextField jTextField8;

// End of variables declaration//GEN-END:variables
}

```

```

package bd;

import java.awt.*;

import java.sql.*;

import java.util.ArrayList;

import javax.swing.*;

import javax.swing.table.*;

public class Worker extends javax.swing.JFrame {

    public Worker() {

        initComponents();

        this.setLocationRelativeTo(null);

        DisplayTable();

    }

    public ArrayList<SetterWorker> tableList(){

        ArrayList<SetterWorker> tableList = new ArrayList<>();

        Connection con = (new GetConnection()).getConnection();

        String query = "select * from worker";

        Statement st;

        ResultSet rs;

        try{

            st = con.createStatement();

            rs = st.executeQuery(query);

            SetterWorker setter;

            while(rs.next()){

                setter = new SetterWorker(rs.getString("wrk_AT"), rs.getString("wrk_name"), rs.getString("wrk_lname"), rs.getFloat("wrk_salary"), rs.getInt("wrk_br_code"));

                tableList.add(setter);

            }

        }catch(Exception ex){

            ex.printStackTrace();

        }

        return tableList;
    }

```

```
}
```

```
private void DisplayTable(){  
  
    ArrayList<SetterWorker> list = tableList();  
  
    DefaultTableModel model = (DefaultTableModel)jTable1.getModel();  
  
    Object [] row = new Object [5];  
  
    for(int i=0; i<list.size(); i++){  
  
        row[0]= list.get(i).getAT();  
  
        row[1]= list.get(i).getName();  
  
        row[2]= list.get(i).getLastName();  
  
        row[3]= list.get(i).getSalary();  
  
        row[4]= list.get(i).getCode();  
  
  
        model.addRow(row);  
  
    }  
}
```

```
public void executeQuery(String query, String message){  
  
    Connection con = (new GetConnection()).getConnection();  
  
    Statement st;  
  
  
  
    try{  
  
        st = con.createStatement();  
  
        if((st.executeUpdate(query))==1){  
  
            DefaultTableModel model = (DefaultTableModel)jTable1.getModel();  
  
            model.setRowCount(0);  
  
            DisplayTable();  
  
            JOptionPane.showMessageDialog(null, "Data "+message+"Successfully");  
  
        }else{  
  
            JOptionPane.showMessageDialog(null, "Data Not"+message);  
  
        }  
  
    }catch(Exception ex){  
  
        ex.printStackTrace();  
  
        JOptionPane.showMessageDialog(null,ex.getMessage() );  
  
    }  
  
}
```



```

@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();
    jLabel4 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    jTextField1 = new javax.swing.JTextField();
    jTextField2 = new javax.swing.JTextField();
    jTextField3 = new javax.swing.JTextField();
    jTextField4 = new javax.swing.JTextField();
    jTextField5 = new javax.swing.JTextField();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();
    jButton3 = new javax.swing.JButton();
    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    jButton4 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);

    jLabel1.setText("wrk_AT");

    jLabel2.setText("wrk_name");

    jLabel3.setText("wrk_lanme");

    jLabel4.setText("wrk_salary");

    jLabel5.setText("wrk_br_code");

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

```

```

});

jButton2.setText("UPDATE");

jButton2.addActionListener(new java.awt.event.ActionListener() {

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

        jButton2ActionPerformed(evt);

    }

});

jButton3.setText("DELETE");

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

        jButton3ActionPerformed(evt);

    }

});

jTable1.setModel(new javax.swing.table.DefaultTableModel(

    new Object [][] {

        },

    new String [] {

        "wrk_AT", "wrk_name", "wrk_lname", "wrk_salary", "wrk_br_code"

    }

) {

    boolean[] canEdit = new boolean [] {

        false, false, false, false, false

    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {

        return canEdit [columnIndex];

    }

});

jTable1.addMouseListener(new java.awt.event.MouseAdapter() {

    public void mouseClicked(java.awt.event.MouseEvent evt) {

        jTable1MouseClicked(evt);

    }

});

jScrollPane1.setViewportViewView(jTable1);

```

```

jButton4.setText("Menu");

jButton4.addActionListener(new java.awt.event.ActionListener() {

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

        jButton4ActionPerformed(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()

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

                .addGroup(layout.createSequentialGroup()

                    .addGroup(layout.createSequentialGroup()

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

                            .addGroup(layout.createSequentialGroup()

                                .addGap(51, 51, 51)

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

                                    .addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED_SIZE, 67, javax.swing.GroupLayout.PREFERRED_SIZE)

                                    .addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED_SIZE, 79, javax.swing.GroupLayout.PREFERRED_SIZE)

                                    .addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED_SIZE, 67, javax.swing.GroupLayout.PREFERRED_SIZE)

                                    .addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED_SIZE, 67, javax.swing.GroupLayout.PREFERRED_SIZE)

                                    .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 67, javax.swing.GroupLayout.PREFERRED_SIZE))

                                .addGap(18, 18, 18)

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

                                    .addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT_SIZE, 116, Short.MAX_VALUE)

                                    .addComponent(jTextField2)

                                    .addComponent(jTextField3)

                                    .addComponent(jTextField4)

                                    .addComponent(jTextField5)))

                                .addGroup(layout.createSequentialGroup()

                                    .addGap(18, 18, 18)

                                    .addComponent(jButton1)

                                    .addGap(18, 18, 18)

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

                                        .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                                        .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

                                    .addGap(18, 18, 18)

                                    .addComponent(jButton3)))

                                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 85, Short.MAX_VALUE)

                                .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 511, javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addGap(21, 21, 21))

);

layout.setVerticalGroup(

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

        .addGroup(layout.createSequentialGroup())

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

                .addGroup(layout.createSequentialGroup())

                    .addGap(40, 40, 40)

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

                        .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addComponent(jLabel1))

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

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

                        .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addComponent(jLabel2))

                    .addGap(15, 15, 15)

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

                        .addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addComponent(jLabel3))

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

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

                        .addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addComponent(jLabel4))

                    .addGap(15, 15, 15)

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

                        .addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addComponent(jLabel5))

                    .addGap(66, 66, 66)

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

                        .addComponent(jButton1)

                        .addComponent(jButton2)

                        .addComponent(jButton3))

                    .addGap(29, 29, 29)

                    .addComponent(jButton4))

            .addGroup(layout.createSequentialGroup())

                .addGap(20, 20, 20)

                .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addContainerGap(25, Short.MAX_VALUE))

);

```

```

        pack();

    } // </editor-fold> // GEN-END: initComponents

    private void jTable1MouseClicked(java.awt.event.MouseEvent evt) // GEN-FIRST:event_jTable1MouseClicked

        DefaultTableModel model = (DefaultTableModel) jTable1.getModel();

        int selectedRowIndex = jTable1.getSelectedRow();

        jTextField1.setText(model.getValueAt(selectedRowIndex, 0).toString());
        jTextField2.setText(model.getValueAt(selectedRowIndex, 1).toString());
        jTextField3.setText(model.getValueAt(selectedRowIndex, 2).toString());
        jTextField4.setText(model.getValueAt(selectedRowIndex, 3).toString());
        jTextField5.setText(model.getValueAt(selectedRowIndex, 4).toString());

    } // GEN-LAST:event_jTable1MouseClicked

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton1ActionPerformed

        String query = "INSERT INTO worker VALUES ('"+jTextField1.getText()+"', '"+jTextField2.getText()+"', '"+jTextField3.getText()+"', '"+jTextField4.getText()+"', '"+jTextField5.getText()+"')";

        executeQuery(query, "Inserted");

    } // GEN-LAST:event_jButton1ActionPerformed

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton2ActionPerformed

        String query = "UPDATE worker SET
        wrk_name='"+jTextField2.getText()+"'+'", wrk_lname='"+jTextField3.getText()+"'+'", wrk_salary='"+jTextField4.getText()+"', wrk_br_code='"+jTextField5.getText()+"' WHERE
        wrk_AT='"+jTextField1.getText()+"'+'";

        executeQuery(query, "Updated");

    } // GEN-LAST:event_jButton2ActionPerformed

    private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton3ActionPerformed

        String query = "call deleteAdmin('"+jTextField2.getText()+"', '"+jTextField3.getText()+"')";

        executeQuery(query, "Delete");

    } // GEN-LAST:event_jButton3ActionPerformed

    private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) // GEN-FIRST:event_jButton4ActionPerformed

        this.setVisible(false);

        new Menu().setVisible(true);    } // GEN-LAST:event_jButton4ActionPerformed

    public static void main(String args[]) {

        java.awt.EventQueue.invokeLater(new Runnable() {

```

```

    public void run() {

        new Worker().setVisible(true);

    }

});

}

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField4;
private javax.swing.JTextField jTextField5;

// End of variables declaration//GEN-END:variables
}

```