**Κλάση DatabaseManager :**

DatabaseManager open() : Συνάρτηση που ενεργοποιεί την επικοινωνία με τη βάση (DatabaseManager).

void close() : Συνάρτηση που απενεργοποιεί την επικοινωνία με τη βάση (DatabaseManager).

Cursor fetchU() : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες του χρήστη. Πιο συγκεκριμένα, περιέχει τα user id, email, κωδικό πρόσβασης και χρηματικό υπόλοιπο του χρήστη.

Cursor fetchS() : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες του καταστήματος. Πιο συγκεκριμένα, περιέχει τα shop id, email, κωδικό πρόσβασης, όνομα, χρηματικό υπόλοιπο, διεύθυνση, πόλη, χωρητικότητα, αριθμό αξιολογήσεων, τύπο κουζίνας, έσοδα, έξοδα, στόχο, αξιολόγηση και τηλέφωνο του καταστήματος.

Cursor fetchR(int sid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με την αξιολόγηση και τον αριθμό αξιολογήσεων ενός καταστήματος με συγκεκριμένο shop id.

Cursor fetchMR(int sid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με την αξιολόγηση και τον αριθμό αξιολογήσεων ενός menu καταστήματος με συγκεκριμένο shop id.

Cursor fetchC() : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες του πελάτη. Πιο συγκεκριμένα, περιέχει τα id, email, κωδικό πρόσβασης, όνομα, χρηματικό υπόλοιπο, πόντους και αριθμό κρατήσεων του χρήστη.

Cursor fetchPrM(int sid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες ενός προϊόντος από το menu καταστήματος με συγκεκριμένο shop id. Πιο συγκεκριμένα, περιέχει τα id, όνομα, κόστος και ποσότητα του προϊόντος.

Cursor fetchPrSD(int id,int sid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες ενός προϊόντος εφοδιασμού από το menu καταστήματος με συγκεκριμένο id προϊόντος και supplier id.

Cursor fetchSP(int sid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με όλα τα προϊόντα εφοδιασμού με συγκεκριμένο supplier id.

Cursor fetchSupH(int id) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με το ιστορικό εφοδιασμού ενός μαγαζιού με συγκεκριμένο id.

Cursor fetchNumRes(int id) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τον αριθμό κρατήσεων ενός πελάτη με συγκεκριμένο id.

Cursor fetchA(int sid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τη πόλη ενός καταστήματος με συγκεκριμένο shop id.

Cursor fetchMPrD(int id) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες ενός προϊόντος menu με συγκεκριμένο id. Ειδικότερα, επιστρέφει τα id, όνομα, κόστος, ποσότητα του προϊόντος.

Cursor fetchRH(int cid,int sid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τo ιστορικό αξιολογήσεων ενός πελάτη για ένα κατάστημα με συγκεκριμένα customer id και shop id.

Cursor fetchRO(int cid,int sid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τo ιστορικό αξιολογήσεων των παραγγελιών ενός πελάτη για ένα κατάστημα με συγκεκριμένα customer id και shop id.

Cursor fetchOH(int sid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις ωράριο λειτουργίας ενός καταστήματος με συγκεκριμένο shop id.

Cursor fetchTables(int sid,int cap) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τα τραπέζια ενός καταστήματος με συγκεκριμένα shop id και χωρητικότητα.

Cursor fetchTD(int tid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες ενός τραπεζιού με συγκεκριμένο table id.

Cursor fetchWD(int wid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες ενός σερβιτόρου με συγκεκριμένο waiter id.

Cursor fetchTR(int tid,String date) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες κρατήσεων ενός τραπεζιού με συγκεκριμένη ημερομηνία και table id.

Cursor fetchNeighT(int tid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τα table id των γειτονικών τραπεζιών ενός τραπεζιού με συγκεκριμένο table id.

Cursor fetchWI(int rid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τα waiter id των σερβιτόρων που είναι ανατεθειμένοι σε ένα reservation με συγκεκριμένο reservation id.

Cursor fetchWaiters(int sid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες των σερβιτόρων ενός μαγαζιού με συγκεκριμένο shop id.

Cursor fetchRD(int rid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες μία κράτησης με συγκεκριμένο reservation id.

Cursor fetchTablesInfo(int sid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες των τραπεζιών ενός καταστήματος με συγκεκριμένο shop id.

Cursor fetchRes(int tid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες των κρατήσεων ενός τραπεζιού με συγκεκριμένο table id.

Cursor fetchCustRes(int cid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες των κρατήσεων ενός πελάτη με συγκεκριμένο customer id.

Cursor fetchCustRec(int cid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες ενός πελάτη με συγκεκριμένο customer id.

Cursor fetchOrder(int cid,int sid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες των παραγγελιών ενός πελάτη για ένα μαγαζί με συγκεκριμένα customer id και shop id.

Cursor fetchOrderD(int id) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες των παραγγελιών ενός πελάτη με συγκεκριμένο customer id.

Cursor fetchInv(int id) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τις πληροφορίες για τη πρόσκληση σε μία δεξίωση με συγκεκριμένο reservation id.

Cursor fetchCN(int id) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με το όνομα ενός πελάτη με συγκεκριμένο customer id.

ArrayList<Integer> fetchFL(int id,int rid) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με τα friend id ενός πελάτη με συγκεκριμένο customer id. Επίσης, περιέχει τους φίλους του χρήστη οι οποίοι δεν είναι καλεσμένοι σε ένα reception με συγκεκριμένο reception id.

Cursor fetchOrderID(int id) : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με το id μίας παραγγελίας ενός πελάτη με συγκεκριμένο customer id και reservation id.

Cursor fetchSupplyID() : fetch() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου Cursor με το max id του εφοδιασμού.

Cursor fetchRA() : fetch() συνάρτηση που επιστρέφει τις πληροφορίες των χώρων δεξίωσης.

Cursor fetchSL() : fetch() συνάρτηση που επιστρέφει τις πληροφορίες των εφοδιασμών.

Cursor fetchRAfromID(int rid) : fetch() συνάρτηση που επιστρέφει το κόστος ενός χώρου δεξίωσης με συγκεκριμένο reception id.

Cursor fetchRec(int rid,String d) : fetch() συνάρτηση που επιστρέφει τις πληροφορίες μίας δεξίωσης για συγκεκριμένο reception id και ημερομηνία.

Cursor fetchAv(int cid,String d) : fetch() συνάρτηση που επιστρέφει τη διαθεσιμότητα των πελατών για μία ημερομηνία d.

Cursor fetchPrO(int id) : fetch() συνάρτηση που επιστρέφει τις πληροφορίες ενός προϊόντος παραγγελίας για συγκεκριμένο order id και ημερομηνία.

Cursor fetchB(int id) : fetch() συνάρτηση που επιστρέφει το χρηματικό υπόλοιπο ενός πελάτη με συγκεκριμένο customer id.

void insertRes(int s\_id, int c\_id, int numofc, String dateR, String time, int t\_id, String address, String requests) : insert() συνάρτηση που εισάγει δεδομένα στα πεδία του πίνακα reservation.

void insertOrder(int s\_id, int c\_id, double cost, String om, String pm,int res\_id) : insert() συνάρτηση που εισάγει δεδομένα στα πεδία του πίνακα c\_order.

void insertSupplyPr(int id, String name, double cost, int s\_id, int q) : insert() συνάρτηση που εισάγει δεδομένα στα πεδία του πίνακα supply\_product.

void insertSupply(int s\_id,int supid, String ad, String s,double cost) :

insert() συνάρτηση που εισάγει δεδομένα στα πεδία του πίνακα supply.

void insertSR(int s\_id, int c\_id,double ev) : insert() συνάρτηση που εισάγει δεδομένα στα πεδία του πίνακα rating.

void insertInv(int cid, int rid,String date) : insert() συνάρτηση που εισάγει δεδομένα στα πεδία των πινάκων invitation και calendar.

void insertAss(int r\_id, int w\_id, ArrayList<Integer> tid,String date,int w) :

insert() συνάρτηση που εισάγει δεδομένα στα πεδία του πίνακα res\_waiter.

void insertOR(int oid, int c\_id, double ev) : insert() συνάρτηση που εισάγει δεδομένα στα πεδία του πίνακα rated\_o.

void insertNotif(int sid, int jid) : insert() συνάρτηση που εισάγει δεδομένα στα πεδία του πίνακα notif.

int insertJO(int sid,String pos,double sal,double exp,String sd,String ed) :

insert() συνάρτηση που εισάγει δεδομένα στα πεδία του πίνακα job\_offer.

void insertR(int id, int g, String date, int car, int ca,int cc) : insert() συνάρτηση που εισάγει δεδομένα στα πεδία του πίνακα reception.

void insertPrO(int id, String name, double cost, int o\_id, int quantity) : insert() συνάρτηση που εισάγει δεδομένα στα πεδία του πίνακα o\_product.

void updatePrMQ(int id, int q) : update() συνάρτηση που ενημερώνει τον πίνακα m\_product.

void updatePSupQ(int id, int q) : update() συνάρτηση που ενημερώνει τον πίνακα supplier\_product.

void updateB(int id, int q) : update() συνάρτηση που ενημερώνει τον πίνακα user.

void updateP(int id,int p) : update() συνάρτηση που ενημερώνει τον πίνακα customer.

int updateCRes(int cid, int num) : update() συνάρτηση που ενημερώνει τον πίνακα customer.

void updateR(int sid,double nrat, int n) : update() συνάρτηση που ενημερώνει τον πίνακα shop.

void updateMR(int sid,double nrat, int n) : update() συνάρτηση που ενημερώνει τον πίνακα menu.

**Κλάση artist :**

String name : Το όνομα του καλλιτέχνη που μπορεί να προσλάβει ένας πελάτης για τη δεξίωση του.

int artist\_id : Το id του καλλιτέχνη.

double cost : Το κόστος του καλλιτέχνη.

String music\_genre : Το είδος της μουσικής που ανήκει ο καλλιτέχνης.

getArtist() : get() συνάρτηση που επιστρέφει το artist\_id.

getMusicGenre() : get() συνάρτηση που επιστρέφει το music\_genre.

getName() : get() συνάρτηση που επιστρέφει το name.

**Κλάση calendar :**

int customer\_id : To id του πελάτη.

int reception\_id : Το id της δεξίωσης που έχει οργανωθεί.

String date : Ένα αλφαριθμητικό ημερομηνίας.

getDate() : get() συνάρτηση που επιστρέφει το date.

**Κλάση catering :**

String name : Το όνομα του catering.

int catering\_id : Το id του catering.

String cuisine\_type : Ο τύπος της κουζίνας που προσφέρει το συγκεκριμένο catering

double cost : Το κόστος του catering.

getCatering() : get() συνάρτηση που επιστρέφει το catering\_id.

getCuisineType() : get() συνάρτηση που επιστρέφει το cuisine\_type.

getName() : get() συνάρτηση που επιστρέφει το name.

**Κλάση customer :**

int points : Οι πόντοι που έχει ο πελάτης.

int num\_of\_reservations : Ο αριθμός των κρατήσεων που έχει κάνει ο πελάτης.

void addPointsRShop(int p) : Αυξάνει τα points κατά p.

void addPointsROrder(int p, order o) : Αυξάνει τα points κατά p για αξιολόγηση που έκανε σε μία παραγγελία order o.

int getNumOfReservations(Context c, int id) : Επιστρέφει ένα num τύπου int που αντιπροσωπεύει τον αριθμό των κρατήσεων που έχει κάνει ένας συγκεκριμένος πελάτης.

int updateNumOfReservations(Context c, int cid, int nor) : Ανανεώνει τον αριθμό κρατήσεων ενός πελάτη με αναγνωριστικό cid. Η μέθοδος χρησιμοποιεί το int nor, που είναι ο αριθμός των κρατήσεων και το cid, διότι είναι ορίσματα της updateCRes().

int getPoints : get() συνάρτηση που επιστρέφει τους πόντους του πελάτη.

ArrayList<customer> getCustomer(Context c) : get() συνάρτηση που επιστρέφει χαρακτηριστικά όπως το customer\_id, το email, το password, το όνομα, το χρηματικό υπόλοιπο, τους πόντους και τον αριθμό των κρατήσεών του πελάτη.

ArrayList<reservation> getReservations (Context c, int cid) : get() συνάρτηση που επιστρέφει χαρακτηριστικά όπως το id, το shop id στο οποίο έγινε η κράτηση, το customer id του πελάτη που έκανε την κράτηση, τον αριθμό των ατόμων στη κράτηση, την ημερομηνία, την ώρα το id του τραπεζιού, την διεύθυνση παραλαβής των πελατών και επιπρόσθετα σχόλια για τη κράτηση.

**Κλάση invitation :**

int reception\_id : Tο id της δεξίωσης.

int customer\_id : Το id του πελάτη.

String date : Ένα αλφαριθμητικό ημερομηνίας.

void createInv(Context c, ArrayList<Integer> cid,int rid,String date) : Δημιουργεί την πρόσκληση που θα περιέχει το id του πελάτη, το id της δεξίωσης και μία ημερομηνία.

**Κλάση job\_offer :**

int shop\_id : To id του καταστήματος.

String position : Αλφαριθμητικό για τη θέση εργασίας.

double salary : Ο μισθός που θα λαμβάνει ο εργαζόμενος για τη θέση εργασίας.

double experience : Το χρονικό διάστημα ενασχόλησης με τη συγκεκριμένη θέση που οφείλει να έχει εργαστεί ο υποψήφιος εργαζόμενος.

String start\_date : Ένα αλφαριθμητικό ημερομηνίας, που αφορά την έναρξη της αγγελίας.

String end\_date : Ένα αλφαριθμητικό ημερομηνίας, που αφορά την λήξη της αγγελίας.

int id : To id της αγγελίας.

double getMedianSalary(ArrayList<job\_offer> jl, String pos) : Επιστρέφει ένα meds τύπου double το οποίο αντιπροσωπεύει τον μέσο μισθό που προσφέρουν οι επιχειρήσεις για μία συγκεκριμένη θέση.

String getPosition() : get() συνάρτηση που επιστρέφει το position.

double getSalary() : get() συνάρτηση που επιστρέφει το salary.

ArrayList<job\_offer> getJO(Context c) : get() συνάρτηση που επιστρέφει μια λίστα από τις αγγελίες.

int createJO(Context c, int sid,String pos,double sal,double exp,String sd,String ed) : Επιστρέφει ένα id τύπου int το οποίο αντιπροσωπεύει τη δημιουργία μία νέας θέσης εργασίας.

**Κλάση main\_lists :**

ArrayList<shop> shop\_list = new ArrayList<shop>() : Λίστα τύπου shop που περιέχει τα καταστήματα.

ArrayList<customer> customer\_list = new ArrayList<customer>() : Λίστα τύπου customer που περιέχει τους πελάτες.

ArrayList<user> user\_list = new ArrayList<user>() : Λίστα τύπου user που περιέχει τους χρήστες.

ArrayList<reservation> res\_list = new ArrayList<reservation>() : Λίστα τύπου reservation που περιέχει τα καταστήματα.

ArrayList<job\_offer> job\_offers = new ArrayList<job\_offer>() : Λίστα τύπου job\_offer που περιέχει τις αγγελίες.

ArrayList<reception\_area> reception\_area\_list = new ArrayList<reception\_area>() : Λίστα τύπου reception\_area που περιέχει τις δεξιώσεις.

ArrayList<catering> catering\_list = new ArrayList<catering>() : Λίστα τύπου catering\_list που περιέχει τα catering.

ArrayList<artist> artist\_list = new ArrayList<artist>() : Λίστα τύπου artist που περιέχει τους καλλιτέχνες.

ArrayList<supplier> supplier\_list = new ArrayList<supplier>() : Λίστα τύπου supplier\_list που περιέχει τους προμηθευτές.

**Κλάση menu :**

int shop\_id : Το id του καταστήματος.

double rating : Δηλώνει την αξιολόγηση του καταστήματος.

int num\_of\_rates : Ο αριθμός των αξιολογήσεων.

ArrayList<product\_menu> getMenu(Context c, int sid) : Επιστρέφει ένα pm τύπου ArrayList τύπου product\_menu το οποίο αντιπροσωπεύει το μενού του καταστήματος.

void changeMenuRating(Context c, int sid, double newr) : Αλλάζει την αξιολόγηση του καταστήματος με βάση το id του καταστήματος (sid) και την καινούρια αξιολόγηση (newr).

**Κλάση notification :**

int customer\_id : To id του πελάτη.

String description : Αλφαριθμητικό για τη ειδοποίηση.

String date : Ένα αλφαριθμητικό ημερομηνίας.

void createNot(Context c, int sid,int jid) : Δημιουργεί την ειδοποίηση με βάση το id του καταστήματος και το id (sid) της αγγελίας (jid).

**Κλάση order :**

int customer\_id : Το id του πελάτη.

int order\_id : Το id της παραγγελίας.

int shop\_id : Το id του καταστήματος.

double cost : Το κόστος της παραγγελίας.

String order\_method : Αλφαριθμητικό που δηλώνει τη μέθοδο παραγγελίας.

String payment\_method : Αλφαριθμητικό που δηλώνει τον τρόπο πληρωμής.

int res\_id : Το id της κράτησης.

int getOrder\_id() : get() συνάρτηση που επιστρέφει το order\_id.

int getShop\_id() : get() συνάρτηση που επιστρέφει το shop\_id.

double getCost() : get() συνάρτηση που επιστρέφει το cost.

ArrayList<order> getOrder(Context c, int cid, int sid) : Επιστρέφει ένα αντικείμενο o τύπου ArrayList τύπου order το οποίο αντιπροσωπεύει την παραγγελία.

int getOrderID(Context c, int id) : Επιστρέφει ένα αντικείμενο o τύπου int το οποίο αντιπροσωπεύει το id της παραγγελίας.

ArrayList<order> getOrderD(Context c, int id) : Επιστρέφει ένα o τύπου ArrayList τύπου order το οποίο αντιπροσωπεύει τις πληροφορίες της παραγγελίας.

int getRes\_id() : Επιστρέφει ένα res\_id τύπου int το οποίο αντιπροσωπεύει το id της κράτησης.

String getOrder\_method() : Επιστρέφει ένα order\_method

τύπου String το οποίο αντιπροσωπεύει τη μέθοδο της παραγγελίας.

String getPayment\_method() : Επιστρέφει ένα payment\_method

τύπου String το οποίο αντιπροσωπεύει τη μέθοδο της πληρωμής.

void createO(Context c, String pm, String om, double tc, int sid, int id, int resid, ArrayList<product\_menu> arrayList, ArrayList<Integer> oq, ArrayList<Integer> mq, ArrayList<Integer> mpr) : : Δημιουργεί την παραγγελία με βάση τη μέθοδο πληρωμής (pm), τη μέθοδο παραγγελίας (om), τη χωρητικότητα του τραπεζιού (tc), το id του καταστήματος (sid), το id της παραγγελίας (id), το id της κράτησης (resid) και τη λίστα μενού (arrayList).

**Κλάση product :**

String name : Αλφαριθμητικό που περιέχει το όνομα του προϊόντος.

double price : Η τιμή ενός προϊόντος.

int id : Το id του προϊόντος.

String getName() : Επιστρέφει ένα name τύπου String το οποίο αντιπροσωπεύει την ονομασία του προϊόντος.

double getPrice() : Επιστρέφει ένα price τύπου double το οποίο αντιπροσωπεύει την τιμή του προϊόντος.

int getId() : Επιστρέφει ένα id τύπου int το οποίο αντιπροσωπεύει το id του προϊόντος.

**Κλάση product\_menu :**

int shop\_id : Το id του καταστήματος.

int available\_quantity : Η διαθέσιμη ποσότητα του προϊόντος.

int getQuantity() : Επιστρέφει ένα available\_quantity τύπου int το οποίο αντιπροσωπεύει την ποσότητα του προϊόν μενού.

int updateQuantity(int q) : Επιστρέφει ένα available\_quantity τύπου int το οποίο ανανεώνει την ποσότητα του προϊόντος μενού, μετά από μια παραγγελία.

ArrayList<product\_menu> getMenuPr(Context c, int sid) : Επιστρέφει ένα pm τύπου ArrayList τύπου product\_menu το οποίο αντιπροσωπεύει τα προϊόντα του μενού, με βάση το κατάστημα (sid).

product\_menu getMPrD(Context c, int id,int sid) : Επιστρέφει ένα pm τύπου product\_menu το οποίο αντιπροσωπεύει τις πληροφορίες για τα προϊόντα του μενού, με βάση το κατάστημα (sid).

void createPM(Context c, ArrayList<product\_menu> arrayList, ArrayList<Integer> mq, ArrayList<Integer> mpr) : Δημιουργεί ένα καινούριο προϊόν μενού.

**Κλάση product\_order :**

int order\_id : To id της παραγγελίας προϊόντος.

int quantity : Η ποσότητα παραγγελίας προϊόντος.

int getQuantity() : Επιστρέφει ένα quantity τύπου int το οποίο αντιπροσωπεύει την ποσότητα του προϊόντος.

ArrayList<product\_order> getOrderD(Context c, int id) : Επιστρέφει ένα αντικείμενο o τύπου ArrayList τύπου product\_order το οποίο αντιπροσωπεύει τις πληροφορίες μια παραγγελίας.

void createPO(Context c, ArrayList<product\_menu> arrayList, int oid, ArrayList<Integer> oq, ArrayList<Integer> mpr) : Δημιουργεί μια παραγγελία ενός προϊόντος.

**Κλάση product\_supplier :**

int supplier\_id : To id του προϊόντος προμηθευτή.

int available\_quantity : Η διαθέσιμη ποσότητα προϊόντος.

int getQuantity() : Επιστρέφει ένα available\_quantity τύπου int το οποίο αντιπροσωπεύει την διαθέσιμη ποσότητα του προϊόντος.

**Κλάση product\_supply :**

int supply\_id : To id της προμήθειας.

int quantity : Η ποσότητα προμήθειας.

void createPS(int pid, String n, double p, int q, ArrayList<Integer> spr, ArrayList<Integer> sq) : Δημιουργεί ένα νέο προϊόν εφοδιασμού για ένα κατάστημα με βάση το id του προϊόντος(pid), το επιλεγμένο προϊόν του προμηθευτή (n), τη τιμή (p), τη ποσότητα (q), τη λίστα με τα ids των επιλεγμένων προϊόντων (spr) και την επιλεγμένη ποσότητα (sq).

void createSupplyPr(Context c, ArrayList<product\_supplier> arrayList, int supid, ArrayList<Integer> sq, ArrayList<Integer> spr) : Εκτελεί μία εντολή insert για συγκεκριμένο προϊόν εφοδιασμού με ορισμένο id, name, price, supply id και quantity.

**Κλάση rating :**

int shop\_id : To id του καταστήματος.

int customer\_id : Το id του πελάτη.

double evaluation : Αξιολόγηση σε μορφή double.

int getsid() : Επιστρέφει ένα shop\_id τύπου int το οποίο αντιπροσωπεύει το id του καταστήματος.

void createRating(Context c, int s\_id, int c\_id, double ev, double sev, int oid, int ok) : Δημιουργεί μια νέα αξιολόγηση για ένα κατάστημα με βάση το id του καταστήματος (s\_id), το id του πελάτη (c\_id) και την αξιολόγηση (ev).

**Κλάση reception :**

int guest\_number : O αριθμός των καλεσμένων.

int reception\_id : Το id της δεξίωσης.

int customer\_id : Το id του πελάτη.

String date : Αλφαριθμητικό για την ημερομηνία.

int artist\_id : Το id του καλλιτέχνης.

int reception\_area\_id : Το id του χώρου δεξίωσης.

int catering\_id : Το id του catering.

int getReception\_id() : Επιστρέφει ένα reception\_id τύπου int το οποίο αντιπροσωπεύει το id της δεξίωσης.

String getDate() : Επιστρέφει ένα date τύπου String το οποίο αντιπροσωπεύει την ημερομηνία της δεξίωσης.

int getInv(Context c, int rid) : Επιστρέφει ένα av τύπου int

**Κλάση reception\_area :**

String name : Αλφαριθμητικό για το όνομα του χώρου δεξίωσης.

int reception\_area\_id : Το id του χώρου δεξίωσης.

int num\_of\_guests : Ο αριθμός των καλεσμένων στην δεξίωση.

double cost : Το ποσό που απαιτεί ο χώρος δεξίωσης.

int getAvailability(Context c, int n, String d,int rid,int num) : Επιστρέφει την διαθεσιμότητα του χώρου δεξίωσης με βάση το όνομα του χώρου (n), το id του χώρου δεξίωσης (rid) και του αριθμού των καλεσμένων (num).

String getName() : Επιστρέφει ένα name τύπου String το οποίο αντιπροσωπεύει την ονομασία του χώρου δεξίωσης.

int getNum\_of\_guests() : Επιστρέφει ένα num\_of\_guests τύπου int το οποίο αντιπροσωπεύει τον αριθμό των καλεσμένων.

int getReception\_area\_id() : Επιστρέφει ένα reception\_area\_id τύπου int το οποίο αντιπροσωπεύει το id του χώρου δεξίωσης.

double getCost() : Επιστρέφει ένα cost τύπου double το οποίο αντιπροσωπεύει το κόστος για το χώρο δεξίωσης.

**Κλάση reservation :**

int shop\_id : Το id του καταστήματος.

int customer\_id : Το id του πελάτη.

reservation\_id : Το id της κράτησης.

num\_of\_customers : Ο αριθμός των πελατών για την κράτηση.

String date : Αλφαριθμητικό για την ημερομηνία της κράτησης.

String time : Αλφαριθμητικό για την ώρα της κράτησης.

int table\_id : Το id του τραπεζιού.

String pick\_up\_address : Αλφαριθμητικό για τη διεύθυνση παραλαβής των πελατών για την μεταφορά στο κατάστημα.

String requests : Αλφαριθμητικό για τα αιτήματα που μπορεί να έχουν οι πελάτες κατά την κράτηση.

String getTime(String d) : Επιστρέφει ένα el ή timer τύπου String το οποίο αντιπροσωπεύει την ημερομηνία.

void createRes(Context c, int sid, int id, int noc, String dateb, String tor, int tid, String address, String req) : Δημιουργεί μια κράτηση με βάση το id του καταστήματος (sid), τον αριθμό των πελατών (noc), την ημερομηνία (dateb), την ώρα κράτησης(tor), το id του τραπεζιού (tid), τη διεύθυνση του πελάτη για την παραλαβή (address) και τα αιτήματα τους (req).

int getReservation\_id() : Επιστρέφει ένα reservation\_id τύπου int το οποίο αντιπροσωπεύει το id της κράτησης.

int getShop\_id() : Επιστρέφει ένα shop\_id τύπου int το οποίο αντιπροσωπεύει το id του καταστήματος.

ArrayList<reservation> getReservation(Context c,int cid) : Επιστρέφει ένα r τύπου ArrayList τύπου reservation με βάση το id του πελάτη.

reservation getRD(Context c, int rid) : Επιστρέφει ένα t τύπου reservation με βάση το id της κράτησης.

int getNum\_of\_customers() : Επιστρέφει ένα num\_of\_customers τύπου int το οποίο αντιπροσωπεύει τον αριθμό των πελατών για την κράτηση.

String getRTime() : Επιστρέφει ένα timer τύπου String το οποίο αντιπροσωπεύει την ώρα της κράτησης.

int getWaiterId(Context c, int rid) : Επιστρέφει ένα r τύπου int που αντιπροσωπεύει το id του σερβιτόρου με βάση το id της κράτησης (rid).

**Κλάση shop (Υποκλάση της user) :**

String address : Αλφαριθμητικό που αφορά την διεύθυνση του καταστήματος.

String city : Αλφαριθμητικό που αφορά την πόλη του καταστήματος.

int capacity : Η χωρητικότητα του καταστήματος.

int numofrates : Ο αριθμός των αξιολογήσεων του καταστήματος.

String cuisine\_type : Αλφαριθμητικό που αφορά το είδος κουζίνας του καταστήματος.

double income : Τα έσοδα του καταστήματος.

double expenses : Τα έξοδα του καταστήματος.

double goal : Ο στόχος που έχει θέση το κατάστημα σε ότι αφορά τα κέρδη.

double rating : Η αξιολόγηση που έχει το κατάστημα.

String phone : Το τηλέφωνο που έχει το κατάστημα.

String getPhone() : get() συνάρτηση που επιστρέφει το τηλέφωνο ενός συγκεκριμένου μαγαζιού.

ArrayList<product\_menu> getShop\_m(Context c,int sid) : get() συνάρτηση που επιστρέφει μία λίστα τύπου product\_menu, που περιέχει όλα τα προϊόντα στο μενού ενός μαγαζιού, δηλαδή για συγκεκριμένο shop id. Ειδικότερα, επιστρέφει id, name, cost, shop id και quantity.

ArrayList<String> getOpeningHours(Context c, int sid) : get() συνάρτηση που επιστρέφει μία λίστα τύπου String, που περιέχει το ωράριο ενός συγκεκριμένου μαγαζιού.

ArrayList<table> getTables(Context c, int sid,int cap, String date, String time) : get() συνάρτηση που επιστρέφει μία λίστα τύπου table, που περιέχει τις πληροφορίες των τραπεζιών για συγκεκριμένο shop id, capacity, ημερομηνία και ώρα.

String getAddress() : get() συνάρτηση που επιστρέφει την διεύθυνση ενός μαγαζιού.

String getCity() : get() συνάρτηση που επιστρέφει την πόλη ενός μαγαζιού.

double[] getInOutGoal(Context c, int id) : get() συνάρτηση που επιστρέφει ένα array τύπου double που περιέχει τα έσοδα, τα έξοδα και το στόχο ενός συγκεκριμένου μαγαζιού.

void changeRating(Context c, int sid, double newr) : Συνάρτηση που εκτελεί μια εντολή update, η οποία μεταβάλλει το rating ενός μαγαζιού με συγκεκριμένο shop id κατά έναν αριθμό newr.

int getNumofrates() : get() συνάρτηση που επιστρέφει τον αριθμό των αξιολογήσεων ενός συγκεκριμένου μαγαζιού.

double getRating() : get() συνάρτηση που επιστρέφει την αξιολόγηση ενός μαγαζιού.

ArrayList<shop> getShops(Context c) : get() συνάρτηση που επιστρέφει μία λίστα τύπου shop, που περιέχει τις πληροφορίες των καταστημάτων.

ArrayList<table> getOpeningHoursAndTables(Context c,int id, int val) : get() συνάρτηση που επιστρέφει μία λίστα τύπου table, που περιέχει τα ωράρια ενός καταστήματος και τις πληροφορίες για τα τραπέζια.

**Κλάση supplier** :

int supplier\_id : Το id του προμηθευτή.

String name : Το όνομα του προμηθευτή.

String getName() : get() συνάρτηση που επιστρέφει το όνομα του προμηθευτή.

int getId() : get() συνάρτηση που επιστρέφει το id του προμηθευτή.

ArrayList<supplier> getSupplierList(Context c) : get() συνάρτηση που επιστρέφει μία λίστα τύπου supplier, που περιέχει τα ονόματα και τα id των προμηθευτών.

ArrayList<product\_supplier> getSupplierProducts(Context c, int sid) :

get() συνάρτηση που επιστρέφει μία λίστα τύπου product\_supplier, που περιέχει τα id των προϊόντων του προμηθευτή για ένα δεδομένο supplier id.

**Κλάση supply** :

int supply\_id : Το id της προμήθειας.

int shop\_id : Το id του καταστήματος.

int supplier\_id : Το id του προμηθευτή.

String address : Αλφαριθμητικό που αφορά τη διεύθυνση αποστολή δείγματος στο κατάστημα.

String sample : Αλφαριθμητικό που αφορά το δείγμα για ένα προϊόν.

double cost : Το κόστος της προμήθειας.

ArrayList<supply> getSupplyHistory(Context c, int id) : Επιστρέφει το ιστορικό προμήθειας, που είναι μία λίστα αντικειμένων τύπου supply με βάση το id του καταστήματος (id).

int getSupplier\_id() : Επιστρέφει τo id του προμηθευτή (supplier\_id) που είναι τύπου int.

void createSupply(Context c, int shid, int supid, String address, String sample, double cost, ArrayList<product\_supplier> arrayList, ArrayList<Integer> sq,ArrayList<Integer> spr) : Δημιουργεί προμήθεια με βάση το id του καταστήματος (shid), το id προμηθευτή (supid), τη διεύθυνση (address), το δείγμα (sample), το κόστος (cost), την ποσότητα της προμήθειας (sq) και το προϊόν προμήθειας (spr).

**Κλάση table** :

int table\_id : Το id του τραπεζιού.

int capacity : Η χωρητικότητα του τραπεζιού.

int shop\_id : Το id του καταστήματος.

int getCapacity() : get() συνάρτηση που επιστρέφει τη χωρητικότητα ενός τραπεζιού.

int getTable\_id() : get() συνάρτηση που επιστρέφει το id ενός συγκεκριμένου τραπεζιού.

ArrayList<reservation> getReservations(Context c, int tid,String dateb, String tor) : get() συνάρτηση που επιστρέφει μία λίστα τύπου reservation, που περιέχει όλες τις κρατήσεις για συγκεκριμένο table id, ημερομηνία και ώρα και χρησιμοποιείται για τη πραγματοποίηση κράτησης.

table getTD(Context c, int tid) : get() συνάρτηση που επιστρέφει ένα αντικείμενο τύπου table, που περιέχει όλες τις πληροφορίες ενός τραπεζιού. Πιο συγκεκριμένα, περιέχει το table id, το capacity και το shop id.

ArrayList <reservation> getReservationsT(Context c, int tid,String date) :

get() συνάρτηση που επιστρέφει μία λίστα τύπου reservation, που περιέχει όλες τις κρατήσεις για συγκεκριμένο table id, ημερομηνία και χρησιμοποιείται για την ανάθεση προσωπικού.

ArrayList<Integer> getNeighbouringTables(Context c, int tid,String date) :

get() συνάρτηση που επιστρέφει μία λίστα τύπου Integer, που περιέχει όλες τα γειτονικά τραπέζια ενός συγκεκριμένου τραπεζιού, στα οποία δεν έχει ανατεθεί σερβιτόρος.

**Κλάση user** **:**

String email : Το email του χρήστη.

String password : Ο κωδικός του χρήστη.

String name : Το όνομα του χρήστη.

int id : Το id του χρήστη.

double balance : Το χρηματικό υπόλοιπο του χρήστη.

int getId() : get() συνάρτηση που επιστρέφει το id του user.

String getEmail() : get() συνάρτηση που επιστρέφει το email του user.

String getPassword() : get() συνάρτηση που επιστρέφει τον κωδικό πρόσβασης του user.

String getName() : get() συνάρτηση που επιστρέφει το όνομα του user.

ArrayList<user> getUsers(Context c) : get() συνάρτηση που επιστρέφει μία λίστα τύπου user, που περιέχει όλους τους χρήστες. Πιο συγκεκριμένα, περιέχει το id, το email, τον κωδικό πρόσβασης, το όνομα και το χρηματικό υπόλοιπο του χρήστη.

ArrayList<Integer> getReservationsC(Context c,int id) : get() συνάρτηση που επιστρέφει μία λίστα τύπου Integer, που περιέχει τον αριθμό των κρατήσεων ενός χρήστη.

double getBalance(Context c,int id) : get() συνάρτηση που επιστρέφει το χρηματικό υπόλοιπο ενός χρήστη.

void setBalance(Context c,int id,double cost) : set() συνάρτηση που μεταβάλλει το χρηματικό υπόλοιπο ενός χρήστη.

**Κλάση waiter :**

String name : Το όνομα του σερβιτόρου.

int waiter\_id : Το id του σερβιτόρου.

int shop\_id : Το id του καταστήματος που εργάζεται ο σερβιτόρος.

ArrayList<waiter> getWaiters(Context c, int sid) : get() συνάρτηση που επιστρέφει μία λίστα τύπου waiter, που περιέχει όλους τους σερβιτόρους ενός συγκεκριμένου μαγαζιού. Πιο συγκεκριμένα, περιέχει το id και το όνομα του σερβιτόρου, αλλά και το id του μαγαζιού στο οποίο δουλεύει.

int getWaiter\_id() : get() συνάρτηση που επιστρέφει το id του σερβιτόρου.

waiter getWD(Context c, int wid) : get() συνάρτηση που έναν αντικείμενο τύπου waiter, που περιέχει όλες τις πληροφορίες ενός συγκεκριμένου σερβιτόρου. Πιο συγκεκριμένα, περιέχει το id και το όνομα του σερβιτόρου, αλλά και το id του μαγαζιού στο οποίο δουλεύει.

String getName() : get() συνάρτηση που επιστρέφει το όνομα του σερβιτόρου.

void createAssessment(Context c, int r\_id,int w\_id,ArrayList<Integer> tid,String date,int w) : Εκτελεί μία εντολή insert για ένα reservation id, ένα waiter id, ένα ArrayList με τα table id, μία ημερομηνία και μία μεταβλητή w που χρησιμοποιείται για να εκτελεστεί συγκεκριμένο κομμάτι κώδικα ανάλογα με το εάν ακολουθείται η βασική ροή ή η εναλλακτική ροή.