**USER STORED PROCEDURES**

CREATE FUNCTION "**spUser\_GetAll**"() RETURNS SETOF "User" AS 'SELECT \* FROM "User";' LANGUAGE 'sql';

CREATE FUNCTION "**spUser\_GetById**"(id integer) RETURNS "User" AS 'SELECT \* FROM "User" WHERE id="ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spUser\_InsertValue**"(username VARCHAR, password VARCHAR, firstname VARCHAR,

middlename VARCHAR, lastname VARCHAR, givenname VARCHAR,

addr1 VARCHAR, addr2 VARCHAR, province VARCHAR, postalcode VARCHAR,

sex VARCHAR, phone VARCHAR, dob TIMESTAMP, email VARCHAR)

RETURNS void AS $$ INSERT INTO "User"("Username", "Password", "FirstName", "MiddleName", "LastName",

"GivenName", "Addr1", "Addr2", "Province", "PostalCode", "Sex", "Phone", "DOB", "Email")

VALUES (username, password, firstname, middlename, lastname, givenname, addr1, addr2, province, postalcode,

sex, phone, dob, email); $$ LANGUAGE 'sql';

CREATE FUNCTION "**spUser\_ModifyById**"(id int, username VARCHAR, password VARCHAR, firstname VARCHAR,

middlename VARCHAR, lastname VARCHAR, givenname VARCHAR, addr1 VARCHAR,

addr2 VARCHAR, province VARCHAR, postalcode VARCHAR, sex VARCHAR,

phone VARCHAR, dob TIMESTAMP, email VARCHAR)

RETURNS void AS $$ UPDATE "User" SET "Username"=username, "Password"=password, "FirstName"=firstname,

"MiddleName"=middlename, "LastName"=lastname, "GivenName"=givenname,

"Addr1"=addr1, "Addr2"=addr2, "Province"=province,

"PostalCode"=postalcode, "Sex"=sex, "Phone"=phone, "DOB" = dob,

"Email" = email WHERE id="ID";$$ LANGUAGE 'sql';

CREATE FUNCTION "**spUser\_DeleteById**"(id int) RETURNS void AS 'DELETE FROM "User" WHERE id="ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spUser\_LastInserted**"(out cur\_user\_id int) AS $$ SELECT "ID" FROM "User" ORDER BY "ID" DESC LIMIT 1; $$ LANGUAGE 'sql';

**CUSTOMER STORED PROCEDURES**

CREATE FUNCTION "**spCustomer\_GetAll**"() RETURNS SETOF "Customer" AS 'SELECT \* FROM "Customer";' LANGUAGE 'sql';

CREATE FUNCTION "**spCustomer\_GetById**"(id integer) RETURNS "Customer" AS 'SELECT \* FROM "Customer" WHERE id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spCustomer\_InsertValue**"(user\_id int, table\_no int) RETURNS void AS 'INSERT INTO "Customer"("User\_ID", "TableNo") VALUES (user\_id, table\_no);' LANGUAGE 'sql';

CREATE FUNCTION "**spCustomer\_Sit**"(user\_id int, tableno int) RETURNS void AS 'UPDATE "Customer" SET

"TableNo" = tableno WHERE user\_id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spCustomer\_Leave**"(user\_id int) RETURNS void AS 'UPDATE "Customer" SET

"TableNo" = null WHERE user\_id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spCustomer\_DeleteById**"(id int) RETURNS void AS 'DELETE FROM “Customer" WHERE id=“User\_ID”;' LANGUAGE 'sql';

CREATE FUNCTION "**spCustomer\_****AtTable**"(tableno int) RETURNS SETOF "Customer" AS 'SELECT \* FROM "Customer"

WHERE "TableNo"=tableno;' LANGUAGE 'sql';

CREATE FUNCTION "**spCustomer\_GetTransactions**"(user\_id int) RETURNS SETOF "Transaction" AS $$ SELECT t."Transaction\_ID", t."Amount", t."Date\_Time" FROM "Customer\_Transaction" AS ct, "Transaction" AS t WHERE ct."User\_ID"=user\_id AND ct."Transaction\_ID"=t."Transaction\_ID"; $$ LANGUAGE 'sql';

CREATE FUNCTION "**spCustomer\_GetReviews**"(user\_id int) RETURNS SETOF "Review" AS $$ SELECT r."User\_ID", r."Review\_ID", r."Description", r."Rating", r."Dish\_ID" FROM "Review" AS r WHERE r."User\_ID"=user\_id; $$ LANGUAGE 'sql';

CREATE FUNCTION "**spCustomer\_GetOrders**"(user\_id int) RETURNS SETOF "Order" AS $$ SELECT \* FROM "Order" AS o WHERE o."User\_ID"=user\_id; $$ LANGUAGE 'sql';

**DISH STORED PROCEDURES**

CREATE FUNCTION "**spDish\_GetAll**"() RETURNS SETOF "Dish" AS 'SELECT \* FROM "Dish";' LANGUAGE 'sql';

CREATE FUNCTION "**spDish\_GetById**"(id integer) RETURNS "Dish" AS 'SELECT \* FROM "Dish" WHERE id="Dish\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spDish\_InsertValue**"(dish\_id int, available boolean, price decimal, description varchar, menu\_type varchar) RETURNS void AS 'INSERT INTO "Dish"("Dish\_ID", "Available", "Price", "Description", "Menu\_Type") VALUES (dish\_id, available, price, description, menu\_type);' LANGUAGE 'sql';

CREATE FUNCTION "**spDish\_ModifyById**"(id int, available boolean, price decimal, description varchar, menu\_type varchar) RETURNS void AS 'UPDATE "Dish" SET "Available" = available,"Price" = price,"Description" = description, "Menu\_Type" = menu\_type WHERE id="Dish\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spDish\_DeleteById**"(id int) RETURNS void AS 'DELETE FROM "Dish" WHERE id="Dish\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spDish\_GetIngredients**"(dish\_id int) RETURNS TABLE(Name varchar) AS $$

SELECT i."Name" FROM "Dish\_Ingredient" AS di, "Ingredient" AS i WHERE di."Dish\_ID"=dish\_id AND di."Ing\_Name"=i."Name"; $$ LANGUAGE 'sql';

CREATE FUNCTION "**spDish\_GetAvailable"()** RETURNS SETOF "Dish" AS $$ SELECT \* FROM "Dish" WHERE "Available"=true; $$ LANGUAGE 'sql';

CREATE FUNCTION "**spDish\_MakeAvailable**"(dish\_id int)RETURNS void AS $$ UPDATE "Dish" SET "Available" = true WHERE dish\_id="Dish\_ID"; $$ LANGUAGE 'sql';

CREATE FUNCTION "**spDish\_MakeUnavailable**"(dish\_id int)RETURNS void AS $$ UPDATE "Dish" SET "Available" = false WHERE dish\_id="Dish\_ID"; $$ LANGUAGE 'sql';

**COOK STORED PROCEDURES**

CREATE FUNCTION "**spCook\_GetAll**"() RETURNS SETOF "Cook" AS 'SELECT \* FROM "Cook";' LANGUAGE 'sql';

CREATE FUNCTION "**spCook\_GetById**"(id integer) RETURNS "Cook" AS 'SELECT \* FROM "Cook" WHERE id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spCook\_InsertValue**"(user\_id int, specialty varchar, type varchar) RETURNS void AS 'INSERT INTO "Cook"("User\_ID", "Specialty", "Type") VALUES (user\_id, specialty, type);' LANGUAGE 'sql';

CREATE FUNCTION "**spCook\_ModifyById**"(user\_id int, specialty varchar, type varchar)RETURNS void AS 'UPDATE "Cook" SET "Specialty" = specialty,"Type" = type WHERE user\_id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spCook\_DeleteById**"(user\_id int) RETURNS void AS 'DELETE FROM "Cook" WHERE user\_id="User\_ID";' LANGUAGE 'sql';

**EMPLOYEE STORED PROCEDURES**

CREATE FUNCTION "**spEmployee\_GetAll**"() RETURNS SETOF "Employee" AS 'SELECT \* FROM "Employee";' LANGUAGE 'sql';

CREATE FUNCTION "**spEmployee\_GetById**"(id integer) RETURNS "Employee" AS 'SELECT \* FROM "Employee" WHERE id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spEmployee\_InsertValue**"(user\_id int, start\_date timestamp, job\_title varchar, salary numeric, mgr\_id int) RETURNS void AS 'INSERT INTO "Employee"("User\_ID", "Start\_Date", "Job\_Title", "Salary", "mgr\_ID") VALUES (user\_id, start\_date, job\_title, salary, mgr\_id);' LANGUAGE 'sql';

CREATE FUNCTION "**spEmployee\_ModifyById**"(user\_id int, start\_date timestamp, job\_title varchar, salary numeric, mgr\_id int)RETURNS void AS 'UPDATE "Employee" SET "Start\_Date" = start\_date,"Job\_Title" = job\_title, "Salary" = salary, "mgr\_ID" = mgr\_id WHERE user\_id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spEmployee\_DeleteById**"(user\_id int) RETURNS void AS 'DELETE FROM "Employee" WHERE user\_id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spEmployee\_****GetManagedBy**"(manager\_id integer) RETURNS SETOF "Employee" AS $$ SELECT \* FROM "Employee" AS e WHERE e."mgr\_ID"=manager\_id; $$ LANGUAGE 'sql';

**MANAGER STORED PROCEDURES**

CREATE FUNCTION "**spManager\_GetAll**"() RETURNS SETOF "Manager" AS 'SELECT \* FROM "Manager";' LANGUAGE 'sql';

CREATE FUNCTION "**spManager\_GetById**"(user\_id integer) RETURNS "Manager" AS 'SELECT \* FROM "Manager" WHERE user\_id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spManager\_InsertValue**"(user\_id int, area varchar) RETURNS void AS 'INSERT INTO "Manager"("User\_ID", "Area") VALUES (user\_id, area);' LANGUAGE 'sql';

CREATE FUNCTION "**spManager\_ModifyById**"(user\_id int, area varchar)RETURNS void AS 'UPDATE "Manager" SET "Area" = area WHERE user\_id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spManager\_DeleteById**"(user\_id int) RETURNS void AS 'DELETE FROM "Manager" WHERE user\_id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spManager\_GetNumberManagers**"(out num\_of\_managers bigint) AS $$ SELECT COUNT(\*) FROM "Manager"; $$ LANGUAGE 'sql';

**WAITER STORED PROCEDURES**

CREATE FUNCTION "**spWaiter\_GetAll**"() RETURNS SETOF "Waiter" AS 'SELECT \* FROM "Waiter";' LANGUAGE 'sql';

CREATE FUNCTION "**spWaiter\_GetById**"(user\_id integer) RETURNS "Waiter" AS 'SELECT \* FROM "Waiter" WHERE user\_id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spWaiter\_InsertValue**"(user\_id int, hours numeric, type varchar) RETURNS void AS 'INSERT INTO "Waiter"("User\_ID", "Hours", "Type") VALUES (user\_id, hours, type);' LANGUAGE 'sql';

CREATE FUNCTION "**spWaiter\_ModifyById**"(user\_id int, hours numeric, type varchar)RETURNS void AS 'UPDATE "Waiter" SET "Hours" = hours, "Type" = type WHERE user\_id="User\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spWaiter\_DeleteById**"(user\_id int) RETURNS void AS 'DELETE FROM "Waiter" WHERE user\_id="User\_ID";' LANGUAGE 'sql';

**INGREDIENT STORED PROCEDURES**

CREATE FUNCTION "**spIngredient\_GetAll**"() RETURNS SETOF "Ingredient" AS 'SELECT \* FROM "Ingredient";' LANGUAGE 'sql';

CREATE FUNCTION "**spIngredient\_GetByName**"(ing\_name varchar) RETURNS "Ingredient" AS 'SELECT \* FROM "Ingredient" WHERE ing\_name="Name";' LANGUAGE 'sql';

CREATE FUNCTION "**spIngredient\_InsertValue**"(ing\_name varchar, price numeric, exp\_date timestamp, quantity numeric) RETURNS void AS 'INSERT INTO "Ingredient"("Name", "Price", "Exp\_Date", "Quantity") VALUES (ing\_name, price, exp\_date, quantity);' LANGUAGE 'sql';

CREATE FUNCTION "**spIngredient\_ModifyByName**"(ing\_name varchar, price numeric, exp\_date timestamp, quantity numeric)RETURNS void AS 'UPDATE "Ingredient" SET "Price" = price, "Exp\_Date" = exp\_date, "Quantity" = quantity WHERE ing\_name="Name";' LANGUAGE 'sql';

CREATE FUNCTION "**spIngredient\_DeleteByName**"(ing\_name varchar) RETURNS void AS 'DELETE FROM "Ingredient" WHERE LOWER(ing\_name)=LOWER("Name");' LANGUAGE 'sql';

**ORDER STORED PROCEDURES**

CREATE FUNCTION "**spOrder\_GetAll**"() RETURNS SETOF "Order" AS 'SELECT \* FROM "Order";' LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_GetById**"(order\_id int) RETURNS "Order" AS 'SELECT \* FROM "Order" WHERE order\_id="Order\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_InsertValue**"(user\_id int, tran\_id int, date\_time timestamp) RETURNS void AS 'INSERT INTO "Order"("User\_ID", "Transaction\_ID", "Date\_Time") VALUES (user\_id, tran\_id, date\_time);' LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_ModifyById**"(order\_id int, user\_id int, tran\_id int, date\_time timestamp)RETURNS void AS 'UPDATE "Order" SET "User\_ID" = user\_id, "Transaction\_ID" = tran\_id,"Date\_Time" = date\_time WHERE order\_id="Order\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_DeleteById**"(order\_id int) RETURNS void AS 'DELETE FROM "Order" WHERE order\_id="Order\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_GetLastInserted**"(out cur\_order\_id int) AS $$ SELECT "Order\_ID" FROM "Order" ORDER BY "Order\_ID" DESC LIMIT 1; $$ LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_NumOrdersByTransaction**"(in tran\_id int, out num\_orders bigint) AS $$ SELECT COUNT("Order\_ID") FROM "Order" WHERE tran\_id="Transaction\_ID"; $$ LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_GetCost**"(in order\_id int, out order\_cost numeric) AS $$ SELECT SUM("Price") FROM "Order\_Dish" AS o,"Dish" AS d WHERE o."Dish\_ID"=d."Dish\_ID" AND o."Order\_ID"=order\_id;

$$ LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_GetDishes**"(order\_id int) RETURNS SETOF "Dish" AS $$ SELECT d."Dish\_ID",d."Available", d."Price", d."Description", d."Menu\_Type" FROM "Dish" AS d, "Order\_Dish" AS od WHERE order\_id=od."Order\_ID" AND d."Dish\_ID"=od."Dish\_ID"; $$ LANGUAGE 'sql';

**TRANSACTION STORED PROCEDURES**

CREATE FUNCTION "**spTransaction\_GetAll**"() RETURNS SETOF "Transaction" AS 'SELECT \* FROM "Transaction";' LANGUAGE 'sql';

CREATE FUNCTION "**spTransaction\_GetById**"(tran\_id int) RETURNS "Transaction" AS 'SELECT \* FROM "Transaction" WHERE tran\_id="Transaction\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spTransaction\_InsertValue**"(amount numeric, date\_time timestamp) RETURNS void AS 'INSERT INTO "Transaction"("Amount", "Date\_Time") VALUES (amount, date\_time);' LANGUAGE 'sql';

CREATE FUNCTION "**spTransaction\_ModifyById**"(tran\_id int, amount numeric, date\_time timestamp)RETURNS void AS 'UPDATE "Transaction" SET "Transaction\_ID" = tran\_id, "Amount" = amount, "Date\_Time" = date\_time WHERE tran\_id="Transaction\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spTransaction\_DeleteById**"(tran\_id int) RETURNS void AS 'DELETE FROM "Transaction" WHERE tran\_id="Transaction\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spTransaction\_GetLastInserted**"(out cur\_tran\_id int) AS $$ SELECT "Transaction\_ID" FROM "Transaction" ORDER BY "Transaction\_ID" DESC LIMIT 1; $$ LANGUAGE 'sql';

CREATE FUNCTION "**spTransaction\_GetAmount**"(in tran\_id int, out tran\_amount numeric) AS $$SELECT "Amount" FROM "Transaction" WHERE tran\_id="Transaction\_ID";$$ LANGUAGE 'sql';

CREATE FUNCTION "**spTransaction\_UpdateAmount**"(tran\_id int, new\_amount numeric) RETURNS void AS $$UPDATE "Transaction" SET "Amount" = new\_amount WHERE tran\_id="Transaction\_ID";$$ LANGUAGE 'sql';

**MENU STORED PROCEDURES**

CREATE FUNCTION "**spMenu\_GetAll**"() RETURNS SETOF "Menu" AS 'SELECT \* FROM "Menu";' LANGUAGE 'sql';

CREATE FUNCTION "**spMenu\_GetByType**"(type varchar) RETURNS "Menu" AS 'SELECT \* FROM "Menu" WHERE type="Type";' LANGUAGE 'sql';

CREATE FUNCTION "**spMenu\_InsertValue**"(type varchar, available boolean) RETURNS void AS 'INSERT INTO "Menu"("Type", "Available") VALUES (type, available);' LANGUAGE 'sql';

CREATE FUNCTION "**spMenu\_ModifyByType**"(type varchar, available boolean)RETURNS void AS 'UPDATE "Menu" SET "Available" = available WHERE type="Type";' LANGUAGE 'sql';

CREATE FUNCTION "**spMenu\_DeleteByType**"(type varchar) RETURNS void AS 'DELETE FROM "Menu" WHERE type="Type";' LANGUAGE 'sql';

CREATE FUNCTION "**spMenu\_GetDishes**"(type varchar) RETURNS SETOF "Dish" AS $$ SELECT d."Dish\_ID",d."Available", d."Price", d."Description", d."Menu\_Type" FROM "Dish" AS d WHERE LOWER(d."Menu\_Type")=LOWER(type); $$ LANGUAGE 'sql';

CREATE FUNCTION "**spMenu\_GetAvailable"()** RETURNS SETOF "Menu" AS $$ SELECT \* FROM "Menu" WHERE "Available"=true; $$ LANGUAGE 'sql';

**REVIEW STORED PROCEDURES**

CREATE FUNCTION "**spReview\_GetAll**"() RETURNS SETOF "Review" AS 'SELECT \* FROM "Review";' LANGUAGE 'sql';

CREATE FUNCTION "**spReview\_GetById**"(user\_id integer, review\_id integer) RETURNS "Review" AS 'SELECT \* FROM "Review" WHERE user\_id="User\_ID" AND review\_id="Review\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spReview\_InsertValue**"(user\_id int, description varchar, rating int, dish\_id int) RETURNS void AS 'INSERT INTO "Review"("User\_ID", "Description", "Rating", "Dish\_ID") VALUES (user\_id, description, rating, dish\_id);' LANGUAGE 'sql';

CREATE FUNCTION "**spReview\_ModifyById**"( user\_id int, review\_id int, description varchar, rating int, dish\_id int)RETURNS void AS 'UPDATE "Review" SET "Description" = description, "Rating" = rating, "Dish\_ID" = dish\_id WHERE user\_id="User\_ID" AND review\_id="Review\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spReview\_DeleteById**"(user\_id int, review\_id int) RETURNS void AS 'DELETE FROM "Review" WHERE user\_id="User\_ID" AND review\_id="Review\_ID";' LANGUAGE 'sql';

**TABLE STORED PROCEDURES**

CREATE FUNCTION "**spTable\_GetAll**"() RETURNS SETOF "Table" AS 'SELECT \* FROM "Table";' LANGUAGE 'sql';

CREATE FUNCTION "**spTable\_GetById**"(tableno int) RETURNS "Table" AS 'SELECT \* FROM "Table" WHERE tableno="TableNo";' LANGUAGE 'sql';

CREATE FUNCTION "**spTable\_InsertValue**"(location varchar, isoccupied boolean, waiter\_id int) RETURNS void AS 'INSERT INTO "Table"("Location", "isOccupied", "waiter\_ID") VALUES (location, isoccupied, waiter\_id);' LANGUAGE 'sql';

CREATE FUNCTION "**spTable\_ModifyById**"(tableno int, location varchar, isoccupied boolean, waiter\_id int)RETURNS void AS 'UPDATE "Table" SET "Location" = location, "isOccupied" = isoccupied, "waiter\_ID" = waiter\_id WHERE tableno="TableNo";' LANGUAGE 'sql';

CREATE FUNCTION "**spTable\_DeleteById**"(tableno int) RETURNS void AS 'DELETE FROM "Table" WHERE tableno ="TableNo";' LANGUAGE 'sql';

CREATE FUNCTION "**spTable\_GetWaitedBy**"(waiter\_id integer) RETURNS SETOF "Table" AS $$ SELECT \* FROM "Table" AS t WHERE t."waiter\_ID"=waiter\_id; $$ LANGUAGE 'sql';

CREATE FUNCTION "**spTable\_MakeOccupied**"(tableno int)RETURNS void AS 'UPDATE "Table" SET "isOccupied" = true WHERE tableno="TableNo";' LANGUAGE 'sql';

CREATE FUNCTION "**spTable\_MakeDisoccupied**"(tableno int)RETURNS void AS 'UPDATE "Table" SET "isOccupied" = false WHERE tableno="TableNo";' LANGUAGE 'sql';

CREATE FUNCTION "**spTable\_****GetOccupied**"() RETURNS SETOF "Table" AS $$ SELECT \* FROM "Table" WHERE "isOccupied"=true; $$ LANGUAGE 'sql';

**DISH\_INGREDIENT STORED PROCEDURES**

CREATE FUNCTION "**spDish\_Ingredient\_GetAll**"() RETURNS SETOF "Dish\_Ingredient" AS 'SELECT \* FROM "Dish\_Ingredient";' LANGUAGE 'sql';

CREATE FUNCTION "**spDish\_Ingredient\_GetById**"(dish\_id int, ing\_name varchar) RETURNS "Dish\_Ingredient" AS 'SELECT \* FROM "Dish\_Ingredient" WHERE dish\_id="Dish\_ID" AND ing\_name="Ing\_Name";' LANGUAGE 'sql';

CREATE FUNCTION "**spDish\_Ingredient\_InsertValue**"(dish\_id int, ing\_name varchar) RETURNS void AS 'INSERT INTO "Dish\_Ingredient"("Dish\_ID", "Ing\_Name") VALUES (dish\_id, ing\_name);' LANGUAGE 'sql';

CREATE FUNCTION "**spDish\_Ingredient\_DeleteById**"(dish\_id int, ing\_name varchar) RETURNS void AS 'DELETE FROM "Dish\_Ingredient" WHERE dish\_id ="Dish\_ID" AND ing\_name="Ing\_Name";' LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_Dish\_getNumberOfDishes**"(in order\_id int, out num\_dishes bigint) RETURNS bigint AS 'SELECT COUNT(\*) FROM "Order\_Dish" WHERE order\_id="Order\_ID"' LANGUAGE 'sql';

**INGREDIENT\_SUPPLIER STORED PROCEDURES**

CREATE FUNCTION "**spIngredient\_Supplier\_GetAll**"() RETURNS SETOF "Ingredient\_Supplier" AS 'SELECT \* FROM "Ingredient\_Supplier";' LANGUAGE 'sql';

CREATE FUNCTION "**spIngredient\_Supplier\_GetById**"(supplier varchar, ing\_name varchar) RETURNS "Ingredient\_Supplier" AS 'SELECT \* FROM "Ingredient\_Supplier" WHERE LOWER(supplier)=LOWER("Supplier") AND LOWER(ing\_name)=LOWER("Ing\_Name");' LANGUAGE 'sql';

CREATE FUNCTION "**spIngredient\_Supplier\_InsertValue**"(supplier varchar, ing\_name varchar) RETURNS void AS 'INSERT INTO "Ingredient\_Supplier"("Supplier", "Ing\_Name") VALUES (supplier, ing\_name);' LANGUAGE 'sql';

CREATE FUNCTION "**spIngredient\_Supplier\_DeleteById**"(supplier varchar, ing\_name varchar) RETURNS void AS 'DELETE FROM "Ingredient\_Supplier" WHERE LOWER(supplier)=LOWER("Supplier") AND LOWER(ing\_name)=LOWER("Ing\_Name");' LANGUAGE 'sql';

CREATE FUNCTION "**spIngredient\_Supplier\_getNumberOfSuppliers**"(in ing\_name varchar, out num\_supplier bigint) RETURNS bigint AS 'SELECT COUNT(\*) FROM "Ingredient\_Supplier" WHERE LOWER(ing\_name)=LOWER("Ing\_Name");' LANGUAGE 'sql';

CREATE FUNCTION "**spIngredient\_Supplier\_****GetIngredientsBySupplier**"(supplier varchar) RETURNS TABLE(Ing\_Name varchar) AS $$ SELECT isup."Ing\_Name" FROM "Ingredient\_Supplier" AS isup WHERE LOWER(isup."Supplier")=LOWER(supplier); $$ LANGUAGE 'sql';

**CUSTOMER\_TRANSACTION STORED PROCEDURES**

CREATE FUNCTION "**spCustomer\_Transaction\_GetAll**"() RETURNS SETOF "Customer\_Transaction" AS 'SELECT \* FROM "Customer\_Transaction";' LANGUAGE 'sql';

CREATE FUNCTION "**spCustomer\_Transaction\_GetById**"(user\_id int, tran\_id int) RETURNS "Customer\_Transaction" AS 'SELECT \* FROM "Csustomer\_Transaction" WHERE user\_id="User\_ID" AND tran\_id="Transaction\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spCustomer\_Transaction\_InsertValue**"(user\_id int, tran\_id int) RETURNS void AS 'INSERT INTO "Customer\_Transaction"("User\_ID", "Transaction\_ID") VALUES (user\_id, tran\_id);' LANGUAGE 'sql';

CREATE FUNCTION "**spCustomer\_Transaction\_DeleteById**"(user\_id int, tran\_id int) RETURNS void AS 'DELETE FROM "Customer\_Transaction" WHERE user\_id="User\_ID" AND tran\_id="Transaction\_ID";' LANGUAGE 'sql';

**ORDER\_DISH STORED PROCEDURES**

CREATE FUNCTION "**spOrder\_Dish\_GetAll**"() RETURNS SETOF "Order\_Dish" AS 'SELECT \* FROM "Order\_Dish";' LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_Dish\_GetById**"(order\_id int, dish\_id int) RETURNS "Order\_Dish" AS 'SELECT \* FROM "Order\_Dish" WHERE order\_id="Order\_ID" AND dish\_id="Dish\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_Dish\_InsertValue**"(order\_id int, dish\_id int) RETURNS void AS 'INSERT INTO "Order\_Dish"("Order\_ID", "Dish\_ID") VALUES (order\_id, dish\_id);' LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_Dish\_DeleteById**"(order\_id int, dish\_id int) RETURNS void AS 'DELETE FROM "Order\_Dish" WHERE order\_id="Order\_ID" AND dish\_id="Dish\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_Ingredient\_NumIngredientInDish**"(in dish\_id int, out num\_ing bigint) RETURNS bigint AS 'SELECT COUNT(\*) FROM "Dish\_Ingredient" WHERE dish\_id="Dish\_ID"' LANGUAGE 'sql';

CREATE FUNCTION "**spOrder\_Dish\_GetOrderList**"(dish\_id int) RETURNS SETOF "Order" AS $$ SELECT "Order\_ID", "User\_ID", "Transaction\_ID", "Date\_Time" FROM "Order\_Dish" AS od NATURAL JOIN "Order" AS o WHERE od."Dish\_ID"=dish\_id; $$ LANGUAGE 'sql';

**ONLINE\_ORDER STORED PROCEDURES**

CREATE FUNCTION "**spOnline\_Order\_GetAll**"() RETURNS SETOF "Online\_Order" AS 'SELECT \* FROM "Online\_Order";' LANGUAGE 'sql';

CREATE FUNCTION "**spOnline\_Order\_GetById**"(order\_id integer) RETURNS "Online\_Order" AS 'SELECT \* FROM "Online\_Order" WHERE order\_id="Order\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spOnline\_Order\_InsertValue**"(order\_id int, application varchar) RETURNS void AS 'INSERT INTO "Online\_Order"("Order\_ID", "Application") VALUES (order\_id, application);' LANGUAGE 'sql';

CREATE FUNCTION "**spOnline\_Order\_ModifyById**"(order\_id int, application varchar)RETURNS void AS 'UPDATE "Online\_Order" SET "Application" = application WHERE order\_id="Order\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spOnline\_Order\_DeleteById**"(order\_id int) RETURNS void AS 'DELETE FROM "Online\_Order" WHERE order\_id="Order\_ID";' LANGUAGE 'sql';

**IN\_STORE\_ORDER STORED PROCEDURES**

CREATE FUNCTION "**spIn\_Store\_Order\_GetAll**"() RETURNS SETOF "In\_Store\_Order" AS 'SELECT \* FROM "In\_Store\_Order";' LANGUAGE 'sql';

CREATE FUNCTION "**spIn\_Store\_Order\_GetById**"(order\_id integer) RETURNS "In\_Store\_Order" AS 'SELECT \* FROM "In\_Store\_Order" WHERE order\_id="Order\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spIn\_Store\_Order\_InsertValue**"(order\_id int, tableno int) RETURNS void AS 'INSERT INTO "In\_Store\_Order"("Order\_ID", "TableNo") VALUES (order\_id, tableno);' LANGUAGE 'sql';

CREATE FUNCTION "**spIn\_Store\_Order\_ModifyById**"(order\_id int, tableno int)RETURNS void AS 'UPDATE "In\_Store\_Order" SET "TableNo" = tableno WHERE order\_id="Order\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spIn\_Store\_Order\_DeleteById**"(order\_id int) RETURNS void AS 'DELETE FROM "In\_Store\_Order" WHERE order\_id="Order\_ID";' LANGUAGE 'sql';

CREATE FUNCTION "**spIn\_Store\_Order\_GetOrdersByWaiter**"(waiter\_id int) RETURNS "In\_Store\_Order" AS $$ SELECT io."Order\_ID", io."TableNo" FROM "In\_Store\_Order" AS io, "Table" AS t WHERE io."TableNo"=t."TableNo" AND t."waiter\_ID"=waiter\_id; $$ LANGUAGE 'sql';

CREATE FUNCTION "**spIn\_Store\_Order\_GetOrdersByTable**"(tableno int) RETURNS "In\_Store\_Order" AS $$ SELECT io."Order\_ID", io."TableNo" FROM "In\_Store\_Order" AS io WHERE io."TableNo"=tableno;

$$ LANGUAGE 'sql';