Reichman University

Adelson School of Entrepreneurship

Final Project

Course: Relational Databases and SQL

Lecturer: Mr. Uri Goldstein

Tutor: Mr. Efraim Pecani

Teaching Assistant: Ms. Mor Menashe

Submit by:

Eylon Shetrit, 316592559

Bar Vanunu, 316251404

Semester A

Date: 26/02/2022

```sql
######################################## Section A - Schema ################################################################

drop database if exists brt;
CREATE DATABASE brt;
use brt;

######################### tags creation ##############################

CREATE TABLE `tags` (
`tag_id` smallint unsigned NOT NULL AUTO_INCREMENT,
`tag` varchar(255) not null,
PRIMARY KEY (`tag_id`)
);

##################### bus_stops creation ############################

CREATE TABLE `bus_stops` (
`station_num` int(255) unsigned NOT NULL unique,
`station_name` varchar(45) NOT NULL,
`address` varchar(90) NOT NULL ,
  PRIMARY KEY (`station_num`)
);

###################### route creation ##############################

CREATE TABLE `route` (
`route_id` smallint unsigned NOT NULL auto_increment,
`line_id` int(2) NOT NULL,
`direction` varchar(15) NOT NULL,
`first_station` int(255) unsigned NOT NULL,
`last_station` int(255) unsigned NOT NULL,
`route_stations` varchar(255)  NOT NULL,
PRIMARY KEY (`route_id`),
FOREIGN KEY (`first_station`) REFERENCES `bus_stops`(station_num),
FOREIGN KEY (`last_station`) REFERENCES `bus_stops`(station_num)
);

###################### Bus2BusStops creation ######################

create TABLE `Bus2BusStops` (
  `bus_id` smallint unsigned NOT NULL,
  `station_num` int(255) unsigned NOT NULL,
  `route_id` smallint unsigned NOT NULL,
  FOREIGN KEY (`station_num`) REFERENCES `bus_stops`(station_num),
  FOREIGN KEY (`route_id`) REFERENCES `route`(route_id)
);

###################### lines creation ##############################

CREATE TABLE `lines` (
`line_id` int(2) NOT NULL AUTO_INCREMENT,
`line` varchar(255) NOT NULL,
`route_up` smallint unsigned NOT NULL,
`route_down` smallint unsigned NOT NULL ,
PRIMARY KEY (`line_id`),
FOREIGN KEY (`route_up`) REFERENCES `route`(route_id),
FOREIGN KEY (`route_down`) REFERENCES `route`(route_id)
);

###################### bus creation ##############################

CREATE TABLE `bus` (
  `bus_id` smallint unsigned NOT NULL AUTO_INCREMENT,
  `licnese_plate` varchar(8) NOT NULL,
  `line_id`  int(2) NOT NULL  ,
  `tag_id` smallint unsigned NOT NULL,
  PRIMARY KEY (`bus_id`),
  FOREIGN KEY (`line_id`) REFERENCES `lines`(line_id),
  FOREIGN KEY (`tag_id`) REFERENCES `tags`(tag_id)
);

############################ tag2bus creation ##################

CREATE TABLE `tag2bus` (
`bus_id` smallint unsigned NOT NULL,
`tag_id` smallint unsigned NOT NULL,
FOREIGN KEY(`bus_id`) REFERENCES `bus`(`bus_id`),
FOREIGN KEY(`tag_id`) REFERENCES `tags`(`tag_id`)
);

############################ capacities creation #####################

CREATE TABLE `capacities` (
`cap_id` int(2) NOT NULL AUTO_INCREMENT,
`tag_id` smallint unsigned NOT NULL,
`name` varchar(20) NOT NULL,
`value` int(3) NOT NULL,
primary key(`cap_id`),
FOREIGN KEY (`tag_id`) REFERENCES `tags`(tag_id)
);

######################################################## Section B - Data ################################################################
######################### tags - Data ##############################

INSERT INTO `brt`.`tags`
(`tag_id`,`tag`)
VALUES
(null,"small"),(null,"medium"),(null,"large"),(null,"extra large"),(null,"double"),
(null,"electric"),(null,"red"),(null,"blue"),(null,"new"),(null,"old");
```
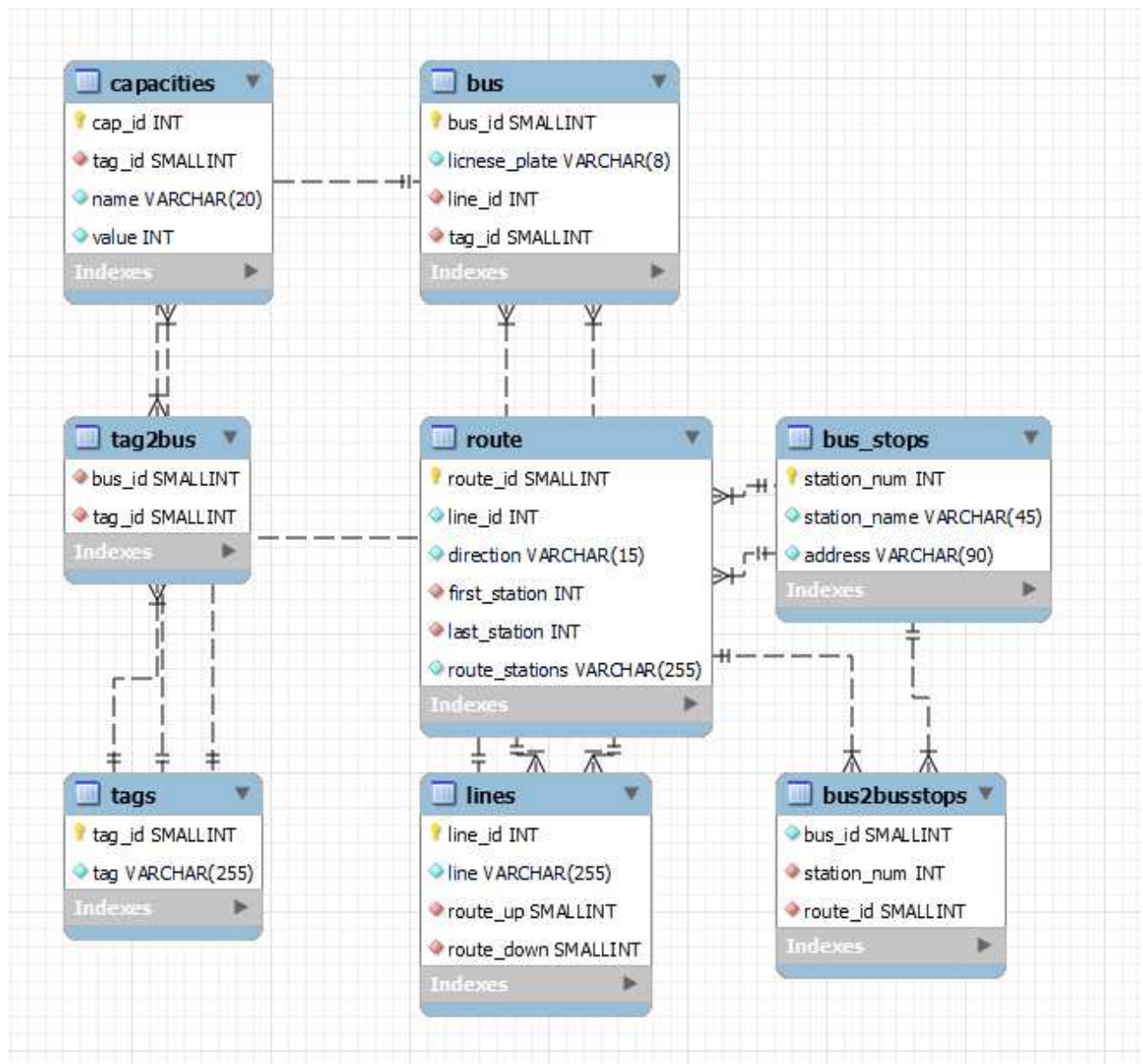
```sql
######################### bus_stops - Data #############################

INSERT INTO `brt`.`bus_stops`
(`station_num`,`station_name`,`address`)
VALUES
(77720,'Reading / docks terminal','Israel Pharmacist'),
(77721,'Hayarkon / Shaar Zion','313 Hayarkon'),
(77722,'Jabotinsky / Ben Yehuda','10 Jabotinsky'),
(77723,'Jabotinsky / Dizengoff','26 Jabotinsky'),
(77724,'Jabotinsky / Four Countries','Jabotinsky'),
(77725,'Jabotinsky / Ibn Gvirol','80 Jabotinsky'),
(77726,'Tel Aviv Municipality / Ibn Gvirol','71 Ibn Gvirol'),
(77727,'Rabbinate of Tel Aviv / King David Blvd','29 King David');

######################### route - Data #############################

INSERT INTO `brt`.`route`
(`route_id`,`line_id`,`direction`,`first_station`,`last_station`,`route_stations`)
VALUES
(null,1,'up',77720,77722,"{'1':'77720','2':'77721','3':'77722'}"),
(null,1,'down',77722,77720,"{'1':'77722','2':'77723','3':'77720'}"),
(null,2,'up',77724,77726,"{'1':'77724','2':'77725','3':'77726'}"),
(null,2,'down',77726,77724,"{'1':'77726','2':'77727','3':'77724'}");
######################### Bus2BusStops - Data #############################

INSERT INTO `brt`.`Bus2BusStops`
(`bus_id`,`station_num`,`route_id`)
VALUES
(1,77720,1),(1,77721,1),(1,77722,1),(2,77720,1),(2,77721,1),(2,77722,1),
(3,77720,1),(3,77721,1),(3,77722,1),(5,77722,2),(5,77723,2),(5,77720,2),
(7,77722,2),(7,77723,2),(7,77720,2),(4,77724,3),(4,77725,3),(4,77726,3),
(6,77724,3),(6,77725,3),(6,77726,3),(8,77726,4),(8,77727,4),(8,77724,4),
(9,77726,4),(9,77727,4),(9,77724,4),(10,77726,4),(10,77727,4),(10,77724,4);

######################### lines - Data #############################

INSERT INTO `brt`.`lines`
(`line_id`,`line`,`route_up`,`route_down`)
VALUES
(null,'55',1,2),
(null,'56',3,4);

######################### bus - Data #############################

INSERT INTO `brt`.`bus`
(`licnese_plate`,`line_id`,`tag_id`)
VALUES
('8822332',1,1),('77234857',1,2),('7564389',1,3),('1236549',2,4),('6542831',1,5),
('7896542',2,1),('74125868',1,2),('78963021',2,3),('753219',2,4),('87956412',2,5);

######################### tag2bus - Data #############################
INSERT INTO `brt`.`tag2bus`
(`bus_id`,`tag_id`)
VALUES
(1,1),(2,2),(3,3),(4,4),(5,5),(6,1),(7,2),(8,3),(9,4),(10,5);

######################### capacities - Data #############################

 INSERT INTO `brt`.`capacities`
(`cap_id`,`tag_id`,`name`,`value`)
VALUES
(null,1,'small',8),(null,2,'medium',16),(null,3,'large',32),(null,4,'extra large',50),(null,5,'double',64);

##################################################### Section C - Queries #############################################################
######################### Query number 1 (2.3.1) #############################
use brt;
select line,
case
when line_id=line_id & first_station = last_station & last_station = first_station & direction != direction then TRUE
else FALSE
end as is_proper_loop
from(
select line,line_id,first_station,last_station,direction
from `lines` as l
join `route` as r using(line_id)
) as A
group by line;

######################### Query number 2 (2.3.2) #############################
use brt;
select b2b.station_num, avg(c.`value`)
from `bus2busstops` as b2b join `bus` using(bus_id)
join capacities as c using(tag_id)
group by b2b.station_num;
```

Section D – Appendix

D.1) BRT‑ entity relationship diagram (ERD)

D.2) Query results

First Query(2.3.1)

line	is_proper_loop
55	1
56	1

Second Query(2.3.2)

station_num	Avrage_Capacity
77720	27.2000
77721	18.6667
77722	27.2000
77723	40.0000
77724	40.8000
77725	29.0000
77726	40.8000
77727	48.6667

D.3) GitHub Repository URL

https://github.com/EylonShetrit/Final_Project_SQL.git

we committed all the files and scripts to this repository, so feel free to pull it from there if you face with an error made by converting scripts to PDF.