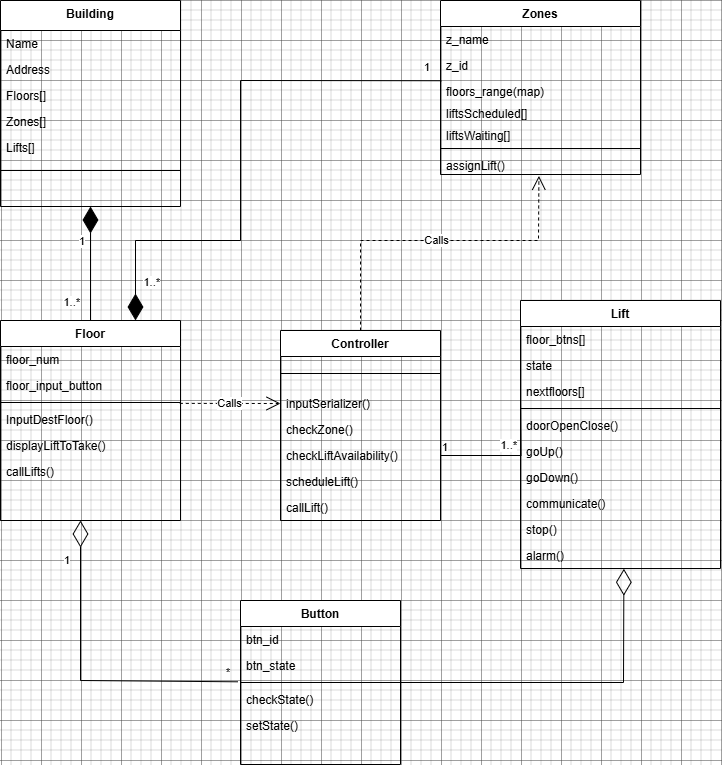
**Assignment: Creating Database for Lift Management System**

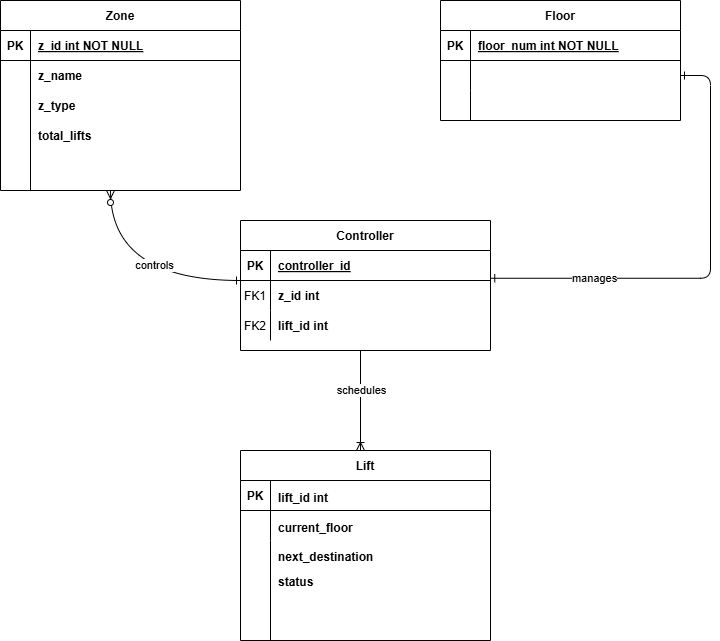
**Asgnmt Coordinator: Suresh Burde**

**Submitted By: Moizuddin A. Shaikh**

**Class Diagram For the Lift Management System**



Entity Realtionship Diagram



**total\_floors**

**Database :** Postgres

**Tables count :** 4

**Tables :** floor, zone, lift, controller

#Creating *zone* table

CREATE TABLE zone (z\_id INT PRIMARY KEY, z\_name varchar(50), z\_type varchar(50), total\_lifts INT);



#Inserting Values

INSERT INTO zone(z\_id, z\_name, z\_type, total\_lifts) VALUES

(1, ‘Zone A’, ‘Retail’, 3),

(2, ‘Zone B’, ‘Offices’, 5),

(3, ‘Zone C’, ‘Residential’, 6),

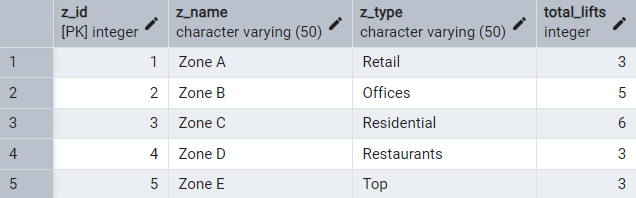
(4, ‘Zone D’, ‘Restaurants’, 3),

(5, ‘Zone E’, ‘Top’, 3);

#View

SELECT \* FROM zone

ORDER BY z\_id ASC



#Alter *zone* Table to add *total\_floors* column on for zone

ALTER TABLE zone

ADD total\_floors INT;

#Adding total floors for all zones in the *total\_floor* column

UPDATE zone

SET total\_floors = CASE

WHEN z\_id = 1 THEN 4

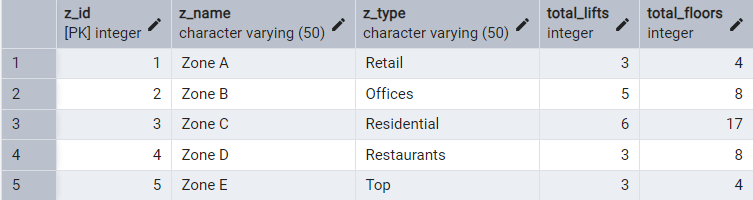
WHEN z\_id = 2 THEN 8

WHEN z\_id = 3 THEN 17

WHEN z\_id = 4 THEN 8

WHEN z\_id = 5 THEN 4

END;



#Creating floor Table

CREATE TABLE floor(floor\_num INT PRIMARY KEY, z\_id INT, FOREIGN KEY(z\_id) REFERENCES zone(z\_id));

INSERT INTO floor(floor\_num, z\_id) VALUES (0,1),(1,1),(2,1),(3,1),(4,2),(5,2),(6,2),(7,2),(8,2),(9,2),(10,2),(11,2),(12,3),(13,3),(14,3),(15,3),(16,3),(17,3),(18,3),(19,3),(20,3),(21,3),(22,3),(23,3),(24,3),(25,3),(26,3),(27,3),(28,3),(29,4),(30,4),(31,4),(32,4),(33,4),(34,4),(35,4),(36,4),(37,5),(38,5),(39,5),(40,5);



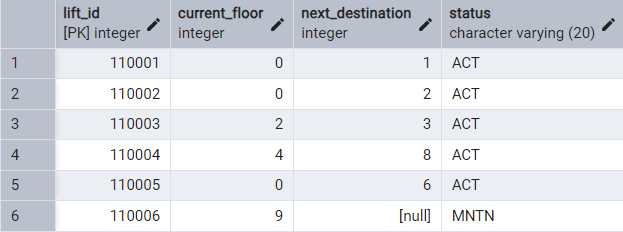
#Creating Lift Table

CREATE TABLE lift(lift\_id INT PRIMARY KEY, current\_floor INT, next\_destination INT, status VARCHAR(20));

#Inserting values in *lift* table

INSERT INTO lift(lift\_id, current\_floor, next\_destination, status) VALUES (110001, 0, 1, 'ACT'), (110002, 0, 2, 'ACT'),

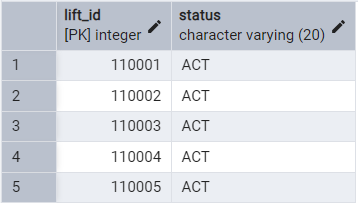
(110003, 2, 3, 'ACT'), (110004, 4, 8, 'ACT'), (110005, 0, 6, 'ACT'), (110006, 9, NULL, 'MNTN');



**#showing active (operational) lifts**

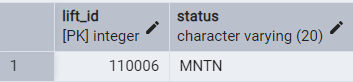
SELECT lift.lift\_id, lift.status FROM lift

WHERE status = 'ACT';



SELECT lift.lift\_id, lift.status FROM lift

WHERE status = 'MNTN';



#from *floor* table counting floors in a zone

*SELECT floor.z\_id, COUNT(floor.floor\_num) FROM floor*

*GROUP BY z\_id*

*ORDER BY z\_id;*

