**Node**

**Relationship schema** : node( x : int , y : int , name : char );

**Create table** :

CREATE TABLE node(

x INT(10) NOT NULL,

y INT(10) NOT NULL,

name CHAR(100) NOT NULL,

PRIMARY KEY(name),

CONSTRAINT uniqueName UNIQUE(x,y)

);

**Insert tuple in node :**

INSERT INTO node (name,x,y) VALUES ('A',5,5);

INSERT INTO node (name,x,y) VALUES ('B',15,5);

INSERT INTO node (name,x,y) VALUES ('C',5,3);

INSERT INTO node (name,x,y) VALUES ('D',11,3);

**Return all node :**  SELECT \* FROM node;

**Return all node where x and y coordinates are same:**

SELECT \* FROM node WHERE x=y;

**Edge**

**Relationship schema :**

edge( edgeName : char , name1 : char, weight : unsigned int default 1 , name2 : char );

--name1 and name2 are foreign key and are from nodes

**Create table edge :**

CREATE TABLE edge(

edgeName CHAR(10) NOT NULL,

name1 CHAR(10) NOT NULL,

weight INT UNSIGNED DEFAULT 1,

name2 CHAR(10) NOT NULL,

PRIMARY KEY(edgeName),

CONSTRAINT uniqueEdge UNIQUE(name1,name2),

FOREIGN KEY(name1) REFERENCES node (name) ON DELETE CASCADE ON UPDATE RESTRICT,

FOREIGN KEY(name2) REFERENCES node (name) ON DELETE CASCADE ON UPDATE RESTRICT

);

**Insert tuple in edge :**

INSERT INTO edge (edgeName,name1,weight,name2) VALUES('edgeAB','A',10,'B');

INSERT INTO edge (edgeName,name1,weight,name2) VALUES('edgeAC','A',2,'C');

INSERT INTO edge (edgeName,name1,weight,name2) VALUES('edgeAD','A',3,'D');

INSERT INTO edge (edgeName,name1,weight,name2) VALUES('edgeDC','D',6,'C');

INSERT INTO edge (edgeName,name1,weight,name2) VALUES('edgeBC','B',7,'C');

**Return all edges :**

SELECT \* FROM edge;

**Update Egde weight :**

UPDATE edge SET weight=20 WHERE edgeName='edgeBC';

**Delete Node on Cascade :**