



```

-- -----reset db----- --
DROP DATABASE IF EXISTS boyeje2_Hockey;
CREATE DATABASE boyeje2_Hockey;
USE boyeje2_Hockey;

-- SOURCE boyeje2_schema.sql;
-- SOURCE boyeje2_data.sql;
-- select * from information_schema.routines where routine_schema='boyeje2_Hockey' \G

-- -----create each table that is required----- --
CREATE OR REPLACE TABLE Person (
  personId INT AUTO_INCREMENT PRIMARY KEY,
  firstName VARCHAR(100) NOT NULL,
  lastName VARCHAR(100) NOT NULL,
  address VARCHAR(100) NOT NULL,
  phone VARCHAR(100) NOT NULL,
  email VARCHAR(100) UNIQUE NOT NULL,
  photo BLOB
) ENGINE=INNODB;

CREATE OR REPLACE TABLE School (
  name VARCHAR(100) PRIMARY KEY,
  address VARCHAR(100) NOT NULL,
  phone VARCHAR(100) NOT NULL,
  email VARCHAR(100) NOT NULL
) ENGINE=INNODB;

CREATE OR REPLACE TABLE Player (
  playerId INT PRIMARY KEY,
  DoB DATE NOT NULL,
  schoolName VARCHAR(100),
  FOREIGN KEY (playerId) REFERENCES Person(personId) ON DELETE CASCADE ON UPDATE CASCADE,
  FOREIGN KEY (schoolName) REFERENCES School(name) ON DELETE RESTRICT ON UPDATE CASCADE
) ENGINE=INNODB;

-- pick list
CREATE OR REPLACE TABLE Qualification (
  name VARCHAR(100) PRIMARY KEY
) ENGINE=INNODB;

CREATE OR REPLACE TABLE Coach (
  coachId INT,
  roleStart DATE NOT NULL,
  FOREIGN KEY (coachId) REFERENCES Person(personId) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=INNODB;

CREATE OR REPLACE TABLE Team (
  name VARCHAR(100) PRIMARY KEY,
  year YEAR NOT NULL,
  ageGroup VARCHAR(100) NOT NULL
) ENGINE=INNODB;

CREATE OR REPLACE TABLE Member (

```

```

shirtNum INT NOT NULL,
teamName VARCHAR(100) NOT NULL,
PRIMARY KEY (teamName, shirtNum),
playerId INT NOT NULL,
FOREIGN KEY (teamName) REFERENCES Team(name) ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (playerId) REFERENCES Player(playerId) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=INNODB;

-- -----joining tables----- --
CREATE OR REPLACE TABLE CoachQualifications (
  coachId INT NOT NULL,
  qualName VARCHAR(100) NOT NULL,
  FOREIGN KEY (coachId) REFERENCES Coach(coachId) ON DELETE CASCADE ON UPDATE CASCADE,
  FOREIGN KEY (qualName) REFERENCES Qualification(name) ON DELETE RESTRICT ON UPDATE
  CASCADE
) ENGINE=INNODB;

CREATE OR REPLACE TABLE TeamCoaches (
  teamName VARCHAR(100) NOT NULL,
  coachId INT NOT NULL,
  FOREIGN KEY (teamName) REFERENCES Team(name) ON DELETE CASCADE ON UPDATE CASCADE,
  FOREIGN KEY (coachId) REFERENCES Coach(coachId) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=INNODB;

CREATE OR REPLACE TABLE GuardianPersons (
  childId INT NOT NULL,
  guardianId INT NOT NULL,
  FOREIGN KEY (childId) REFERENCES Person(personId) ON DELETE CASCADE ON UPDATE CASCADE,
  FOREIGN KEY (guardianId) REFERENCES Person(personId) ON DELETE RESTRICT ON UPDATE CASCADE
) ENGINE=INNODB;

-- -----create function----- --
DELIMITER //
CREATE OR REPLACE FUNCTION teamsize (team varchar(100))
  RETURNS INT
  DETERMINISTIC
  READS SQL DATA
  SQL SECURITY INVOKER
  COMMENT 'This function takes in a team name and will return the number of players in that team'
  BEGIN
    RETURN (SELECT count(shirtNum) FROM Member WHERE teamName = team);
  END //
DELIMITER ;

/*
SELECT teamsize("not a team"); -- returns 0
SELECT teamsize("Southern Icebreakers"); -- returns 3
*/

-- -----create view----- --

-- function to run inside of view
DELIMITER //

```

```

CREATE OR REPLACE FUNCTION bestPhone(childId INT, guardianId INT)
  RETURNS VARCHAR(100)
  DETERMINISTIC
  READS SQL DATA
  SQL SECURITY INVOKER
  COMMENT 'This function returns the guardians phone if it exists, else returns players phone'
  BEGIN
    DECLARE gPhone VARCHAR(100) DEFAULT (SELECT phone FROM Person WHERE personId = guardianId);
    IF gPhone IS NOT NULL THEN RETURN gPhone;
    ELSE RETURN (SELECT phone FROM Person WHERE personId = childId);
    END IF;
    END //
DELIMITER ;
/*
SELECT bestPhone(1, 10); -- returns guardians num
SELECT bestPhone(2, NULL); -- returns childs num
*/

-- playerName, (if guardian), best phone, teamName
CREATE OR REPLACE VIEW phonelist AS
SELECT
  CONCAT(c.firstName, ' ', c.lastName) `Child's Name`,
  CONCAT(g.firstName, ' ', g.lastName) `Guardian's Name`,
  bestPhone(c.personId, g.personId) `Phone`,
  m.teamName `Team Name`
FROM Player p
LEFT JOIN Person c ON p.playerId = c.personId -- c for child
LEFT JOIN GuardianPersons gp ON c.personId = gp.childId
LEFT JOIN Person g ON gp.guardianId = g.personId -- g for guardian
JOIN Member m ON p.playerId = m.playerId;
-- SELECT * FROM phonelist;

```

```

INSERT INTO Person (firstName, lastName, address, phone, email) VALUES
-- players
("Liam", "Thompson", "67 Pohutukawa Drive, Musselburgh, Dunedin 9013", "027-812-3456",
"liam.thompson@email.com"),
("Mia", "Walker", "25 Seaview Road, Opoho, Dunedin 9010", "021-567-8901", "mia.walker@email.com"),
("Sophia", "Miller", "142 Manuka Street, Caversham, Dunedin 9012", "027-890-1234", "sophia.miller@email.com"),
("Ethan", "Robinson", "8 Harbourview Terrace, Ravensbourne, Dunedin 9022", "021-321-6789",
"ethan.robinson@email.com"),
("Charlotte", "Bennett", "34 Bellview Crescent, Kaikorai Valley, Dunedin 9022", "021-987-6543",
"charlotte.bennett@email.com"),

-- parents
("Rachel", "Bennett", "34 Bellview Crescent, Kaikorai Valley, Dunedin 9022", "021-456-7890",
"rachel.bennett@email.com"),
("David", "Bennett", "34 Bellview Crescent, Kaikorai Valley, Dunedin 9022", "022-789-4561",
"david.bennett@email.com"),
("Sarah", "Robinson", "8 Harbourview Terrace, Ravensbourne, Dunedin 9022", "022-678-1234",
"sarah.robinson@email.com"),
("Michael", "Robinson", "15 Fernhill Road, Mornington, Dunedin 9011", "027-345-6789",
"michael.robinson@email.com"),
("Olivia", "Thompson", "54 Pineview Avenue, St Clair, Dunedin 9012", "027-789-1234",
"olivia.thompson@email.com"), -- also coach

-- coaches
("James", "Anderson", "18 Oakridge Road, North Dunedin, Dunedin 9016", "021-654-3210",
"james.anderson@email.com"),
("Emily", "Taylor", "33 Rimu Road, Mornington, Dunedin 9011", "022-345-6789", "emily.taylor@email.com"),
("Liam", "Wilson", "89 Kowhai Street, Roslyn, Dunedin 9010", "021-234-5678", "liam.wilson@email.com"),
("Mason", "Miller", "12 Totara Crescent, Andersons Bay, Dunedin 9013", "027-456-7890",
"sophie.miller@email.com");

-- (childId, guardianId)
INSERT INTO GuardianPersons VALUES
(1, 10), -- thompson, l & o
(4, 8), -- robinson, e & s
(4, 9), -- robinson, e & m
(5, 6), -- bennet, c & r
(5, 7); -- bennet, c & d

-- (name, address, phone, email)
INSERT INTO School VALUES
("Southern Hills Secondary School", "123 Oakridge Road, North Dunedin, Dunedin 9016", "03-467-1123",
"info@southernhills.email.com"),
("Harbour View High School", "56 Seaview Road, Opoho, Dunedin 9010", "03-470-3456",
"contact@harbourviewhigh.email.com"),
("Pinegrove College", "34 Manuka Street, Caversham, Dunedin 9012", "03-455-1234",
"office@pinegrovecollege.email.com"),
("Fernwood Intermediate", "78 Kowhai Street, Roslyn, Dunedin 9010", "03-476-7890",
"admin@fernwoodint.email.com"),
("Westbrook High School", "9 Totara Crescent, Mornington, Dunedin 9011", "03-477-5678",
"enquiries@westbrookhs.email.com");

-- (playerId, DoB, schoolName)

```

```

INSERT INTO Player VALUES
(1, "2006-03-16", "Southern Hills Secondary School"),
(2, "2007-07-23", "Southern Hills Secondary School"),
(3, "2008-11-05", "Harbour View High School"),
(4, "2009-02-10", "Fernwood Intermediate"),
(5, "2008-08-14", NULL);

-- (name)
INSERT INTO Qualification VALUES
("Certified Hockey Performance Coach (CHPC)",
("NZ Hockey Level 1 Coaching Certificate"),
("NZ Hockey Level 2 Coaching Certificate"),
("NZ Hockey Elite Coaching Qualification"),
("National Hockey Development Coaching Diploma");

-- (personId, roleStart, qualification)
INSERT INTO Coach VALUES
(10, "2023-02-20"),
(11, "2020-11-16"),
(12, "2010-12-13"),
(13, "1997-11-28"),
(14, "2014-02-19");

-- (name, year, ageGroup)
INSERT INTO Team VALUES
("Dunedin Frost Giants", "2021", "u18"),
("Southern Icebreakers", "2022", "u15"),
("Dunedin Glaciers", "2020", "u12"),
("Otago Thunderhawks", "2019", "open"),
("Harbour City Huskies", "2023", "u15");

-- (shirtNum, teamName, playerId)
INSERT INTO Member VALUES
(8, "Dunedin Frost Giants", 1),
(23, "Southern Icebreakers", 2),
(17, "Southern Icebreakers", 3),
(11, "Harbour City Huskies", 4),
(42, "Southern Icebreakers", 5);

-- (coachId, qualName)
INSERT INTO CoachQualifications VALUES
(10, "Certified Hockey Performance Coach (CHPC)",
(11, "NZ Hockey Level 1 Coaching Certificate"),
(12, "NZ Hockey Level 2 Coaching Certificate"),
(13, "NZ Hockey Level 1 Coaching Certificate"),
(13, "NZ Hockey Level 2 Coaching Certificate"),
(13, "National Hockey Development Coaching Diploma"),
(14, "NZ Hockey Level 1 Coaching Certificate"),
(14, "NZ Hockey Level 2 Coaching Certificate");

-- (teamName, coachId)
INSERT INTO TeamCoaches VALUES
("Dunedin Frost Giants", 12),
("Southern Icebreakers", 11),

```

("Dunedin Glaciers", 10),
("Dunedin Glaciers", 13),
("Otago Thunderhawks", 14),
("Harbour City Huskies", 12);