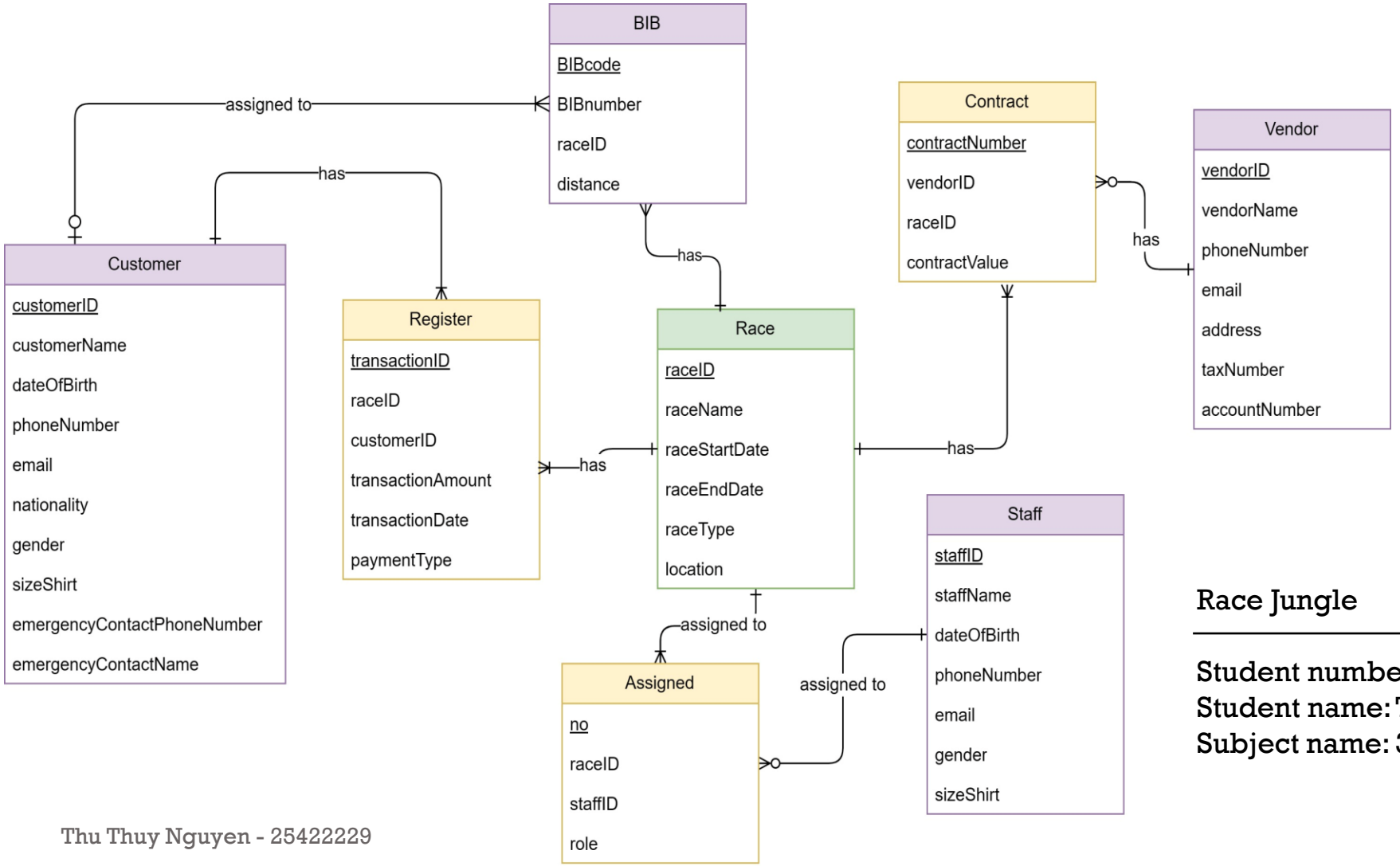
STUDENT NUMBER: 25422229
STUDENT NAME: THU THUY NGUYEN

RACE JUNGLE COMPANY
SUBJECT NAME: 32606, DATABASE – AUTUMN 2024

ABOUT RACE JUNGLE:

- This database demonstrates the Race Jungle website and database.
- Race Jungle was established in 2017 and is one of the first four running race-organized companies in Vietnam.
- The main race series that Race Jungle organized include Ultra Trail, Discovery Marathon, Jungle Paths, Mount Paths, and Fastest X.
- Race Jungle has a database with more than 30.000 runners as customers, 200 staff and almost 100 vendors. This article is a simulated database.
- Race Jungle website: https://racejungle.com/

THE ERD FOR RACE JUNGLE DATABASE:



Race Jungle

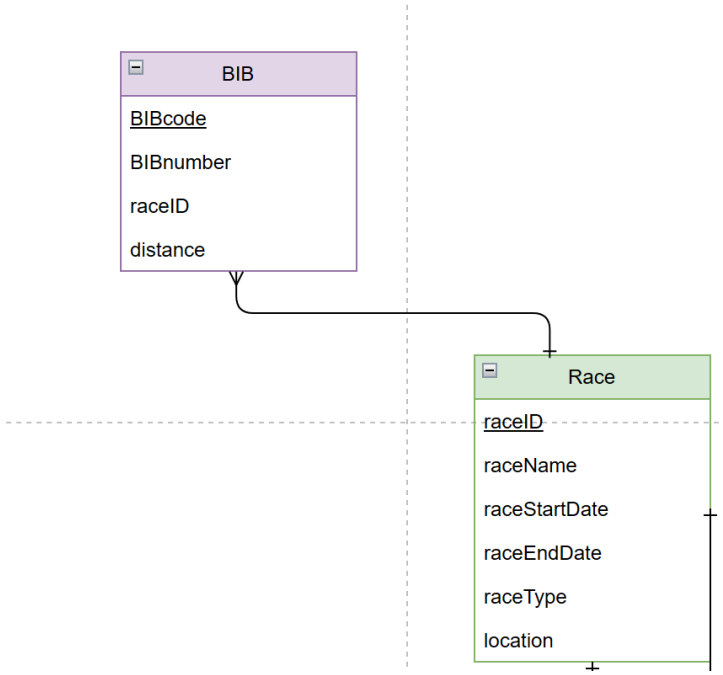
Student number: 25422229
Student name: Thu Thuy Nguyen
Subject name: 32606, Database

A SINGLE ONE-TO-MANY RELATIONSHIP:

One Race can have one to many BIB codes, but a BIB code only belongs to one race.

For example, 'Cuc Phuong Jungle Path 2024' has many BIB codes such as: 875FH4295, 6501G774N…

```
postgres=# select BIB.BIBcode, BIB.BIBnumber, Race.raceName
postgres-# FROM BIB, Race
postgres-# WHERE BIB.raceID = Race.raceID;
  bibcode   | bibnumber |              racename
------------+-----------+-----------------------------------
 875FH4295 |     50008 | Cuc Phuong Jungle Path 2024
 2R702J876 |     50008 | Ha Giang Discovery Marathon 2024
 821Y4K144 |     50100 | Quang Binh Discovery Marathon 2024
 4023JM467 |     10015 | Viettel Fastest X 2024
 3749521UC |     10028 | Ha Giang Discovery Marathon 2024
 48320T56Z |     10123 | Mau Son Mouth Path 2024
 1W368A015 |     25002 | Cao Bang Ultra Trail 2024
 520505XA9 |     25018 | Bac Son Ultra Trail 2024
 QV4609503 |     25018 | Ha Giang Discovery Marathon 2024
 6501G774N |     25032 | Cuc Phuong Jungle Path 2024
 432158D8S |     25106 | Cat Tien Jungle Path 2024
 944F8L981 |     42003 | Cao Bang Ultra Trail 2024
 7E69R9455 |     42105 | Mau Son Mouth Path 2024
 A8S919670 |     42115 | Cuc Phuong Jungle Path 2024
 24F16J170 |     70001 | Ha Giang Discovery Marathon 2024
 12V60H578 |     70080 | Quang Binh Discovery Marathon 2024
```



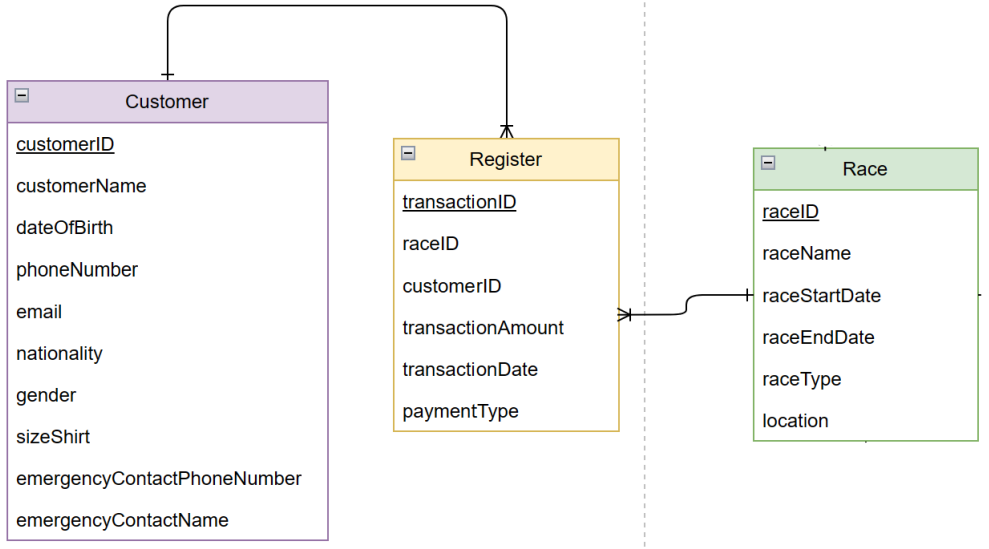
A SINGLE MANY-TO-MANY RELATIONSHIP:

One race can have many customers. One customer can register for more than one race.

For example, customer name 'Trinh Diep Anh' has registered for 2 races 'Ha Giang Discovery Marathon 2024' and 'Bac Son Ultra Trail 2024'.

Race 'Ha Giang Discovery Marathon 2024' has many customers such as: Trinh Diep Anh, Bui Khanh Duong, Luu Minh Nguyet …

```
postgres=# select c.customerName, r.raceName, register.transactionID
postgres-# FROM Customer c, Race r, register
postgres-# WHERE c.customerID = register.customerID
postgres-# AND r.raceID = register.raceID;
   customername   |              racename               | transactionid
------------------+-------------------------------------+---------------
 Tran Ngoc Minh   | Cuc Phuong Jungle Path 2024         | 9X023K856
 Le Hoang Anh     | Cat Tien Jungle Path 2024           | 21D376F27
 Nguyen Phuong Vy | Cao Bang Ultra Trail 2024           | R565R0377
 Dao Hai Nam      | Mau Son Mouth Path 2024             | 83T37811Y
 Phan Nhat Minh   | Cuc Phuong Jungle Path 2024         | R65021G02
 Trinh Diep Anh   | Ha Giang Discovery Marathon 2024    | 2J25494U2
 Luu Minh Nguyet  | Quang Binh Discovery Marathon 2024  | 4BH523872
 Ho Mai Trang     | Viettel Fastest X 2024              | 683UR0256
 Bui Khanh Duong  | Ha Giang Discovery Marathon 2024    | U3827Z906
 Duong Lan Chi    | Mau Son Mouth Path 2024             | 7782446UE
 Phan Nhat Minh   | Cao Bang Ultra Trail 2024           | G3359K646
 Trinh Diep Anh   | Bac Son Ultra Trail 2024            | 4410UY963
 Luu Minh Nguyet  | Ha Giang Discovery Marathon 2024    | 2IO401111
 Cao Thanh Binh   | Quang Binh Discovery Marathon 2024  | 467997U5P
 Duong Tu Anh     | Cat Ba Fastest X 2024               | 3K228G960
 Vu Hoang Long    | Viettel Fastest X 2024              | 502E3Y738
 Miyuki Toshida   | Ha Giang Discovery Marathon 2024    | TY5291400
```



A SIMPLE QUERY OF A SINGLE TABLE:

- List all Customer's IDs, name, size shirts, gender, and nationality who is not from Vietnam:

SELECT customerID, customerName, sizeShirt, gender, nationality

FROM Customer

WHERE nationality NOT LIKE '%Vietnam%';

```
postgres=# SELECT customerID, customerName, sizeShirt, gender, nationality
postgres-# FROM Customer
postgres-# WHERE nationality NOT LIKE '%Vietnam%';
 customerid |  customername  | sizeshirt | gender |          nationality
------------+----------------+-----------+--------+-------------------------------
  809170736 | Qiwei Chen     | XXL       | M      | China
  785747286 | Miyuki Toshida | M         | M      | Japan
```


A QUERY WHICH USES THE WORDS "NATURAL JOIN":

- List the race Name, the Bib number, the distance of race ‘Ha Giang Discovery Marathon 2024’.

SELECT raceName, BIBnumber, distance

FROM race NATURAL JOIN BIB

WHERE raceName = 'Ha Giang Discovery Marathon 2024';

```
postgres=# SELECT raceName, BIBnumber, distance
postgres-# FROM race NATURAL JOIN BIB
postgres-# WHERE raceName = 'Ha Giang Discovery Marathon 2024';
             racename              | bibnumber | distance
-----------------------------------+-----------+----------
 Ha Giang Discovery Marathon 2024 |     50008 | 5
 Ha Giang Discovery Marathon 2024 |     10028 | 10
 Ha Giang Discovery Marathon 2024 |     25018 | 25
 Ha Giang Discovery Marathon 2024 |     70001 | 70
```


THE CROSS PRODUCT EQUIVALENT TO THE "NATURAL JOIN" QUERY ABOVE:

- List the race Name, the Bibnumber, the distance of race ‘Ha Giang Discovery marathon 2024’.

SELECT r.raceName, b.BIBnumber, b.distance

FROM Race r, BIB b

WHERE r.raceID = b.raceID

AND r.raceName = 'Ha Giang Discovery Marathon 2024';

```
postgres=# SELECT r.raceName, b.BIBnumber, b.distance
postgres-# FROM Race r, BIB b
postgres-# WHERE r.raceID = b.raceID
postgres-# AND r.raceName = 'Ha Giang Discovery Marathon 2024';
              racename               | bibnumber | distance
-------------------------------------+-----------+----------
 Ha Giang Discovery Marathon 2024 |     50008 | 5
 Ha Giang Discovery Marathon 2024 |     10028 | 10
 Ha Giang Discovery Marathon 2024 |     25018 | 25
 Ha Giang Discovery Marathon 2024 |     70001 | 70
```


A QUERY INVOLVING A "GROUP BY", ALSO WITH A "HAVING" :

- List the staff who was assigned more than 1 race:

SELECT staffName, staffID, count(staffID) as no
from Staff NATURAL JOIN Assigned

GROUP BY staffID

HAVING count(*) >=2

ORDER BY staffName;

```
postgres=# SELECT staffName, staffID, count(staffID) as No
postgres-# FROM staff NATURAL JOIN assigned
postgres-# GROUP BY staffID
postgres-# HAVING count(staffID) > 1
postgres-# ORDER BY staffName;
    staffname     |  staffid  | no
------------------+-----------+----
 Nguyen Phuong Vy | 802431975 |  2
 Pham Minh Duc    | 756820683 |  2
(2 rows)
```

- List all customer who join more than 1 race:

SELECT customerID, count(*) as No

FROM Register

GROUP BY customerID

HAVING count(*)>1

ORDER BY customerID;

```
postgres=# SELECT customerID, count(*) as No
postgres-# FROM Register
postgres-# GROUP BY customerID
postgres-# HAVING count(*)>1
postgres-# ORDER BY customerID;
 customerid | no
------------+----
  176939411 |  2
  434864165 |  3
  516497348 |  2
(3 rows)
```


A QUERY WHICH USES A SUB QUERY:

- List the vendor who has the most contract value:

SELECT vendorName, contractValue

FROM vendor NATURAL JOIN Contract

WHERE contractValue = (SELECT max(contractValue) FROM Contract);

```
postgres=# SELECT vendorName, contractValue
postgres-# FROM vendor NATURAL JOIN Contract
postgres-# WHERE contractValue = (SELECT max(contractValue) FROM Contract);
      vendorname      | contractvalue
----------------------+---------------
 Viet Phong Logistics |         20000
(1 row)
```


A CROSS PRODUCT WHICH CANNOT BE IMPLEMENTED USING THE WORDS "NATURAL JOIN":

- List all BIB number(s) of two distance 10km and 25km of the race 'HGDM24':

```
SELECT a.BIBnumber as distance10km, b.BIBnumber as distance25km, a.raceID
FROM BIB a, BIB b
WHERE a.raceID = b.raceID
AND a.raceID = 'HGDM24'
AND a.distance = '10'
AND b.distance = '25';
```

```
postgres=# SELECT a.BIBnumber as distance10km, b.BIBnumber as distance25km, a.raceID
postgres-# FROM BIB a, BIB b
postgres-# WHERE a.raceID = b.raceID
postgres-# AND a.raceID = 'HGDM24'
postgres-# AND a.distance = '10'
postgres-# AND b.distance = '25';
 distance10km | distance25km | raceid
--------------+--------------+--------
        10028 |        25018 | HGDM24
(1 row)
```


CHECK STATEMENTS:

```
Create table Register
(transactionID          TEXT            NOT NULL,
raceID                  TEXT,
customerID              integer,
transactionAmount       integer,
transactionDate         date,
paymentType             TEXT,
CONSTRAINT RegisterPK PRIMARY KEY (transactionID),
CONSTRAINT di_table_Register_paymentType CHECK (paymentType IN ('VM', 'BT', 'OP', 'PP')));
```

```
Create table Contract
(contractNumber TEXT    NOT NULL,
vendorID        integer,
raceID          TEXT,
contractValue   integer,
CONSTRAINT ContractPK PRIMARY KEY (contractNumber),
CONSTRAINT di_table_Contract_contractValue CHECK (contractValue > 0));
```


CHECK STATEMENTS:

```
Create table Customer
(customerID                     integer         NOT NULL,
customerName                    TEXT            NOT NULL,
dateOfBirth                     date            NOT NULL,
phoneNumber                     integer         NOT NULL,
email                           TEXT            NOT NULL,
nationality                     char(30),
gender                          char(1),
sizeShirt                       char(3),
emergencyContactPhoneNumber     integer,
emergencyContactName            TEXT,
CONSTRAINT CustomerPK PRIMARY KEY (customerID),
CONSTRAINT di_table_Customer_gender CHECK (gender IN ('M', 'F')),
CONSTRAINT di_table_Customer_email CHECK ((email LIKE '%@%') AND (email LIKE '%.%')),
CONSTRAINT di_table_Customer_nationality CHECK (nationality IN (
     'Afghanistan', 'Akrotiri', 'Albania', 'Algeria', 'Andorra', 'Angola',
     'Antigua and Barbuda', 'Argentina', 'Armenia', 'Aruba', 'Australia',
     'Austria', 'Azerbaijan', 'Bahamas', 'Bahrain', 'Bangladesh', 'Barbados',
```


ON DELETE CASCADE:

```
Create table Staff
(staffID        integer         NOT NULL,
staffName       TEXT            NOT NULL,
dateOfBirth     date,
phoneNumber     integer         NOT NULL,
email           TEXT            NOT NULL,
gender          char(1),
sizeShirt       char(3),
CONSTRAINT StaffPK PRIMARY KEY (staffID),
CONSTRAINT StaffFK FOREIGN KEY (staffID) REFERENCES Assigned
        ON DELETE CASCADE
        ON UPDATE CASCADE,
CONSTRAINT di_table_Staff_gender CHECK (gender IN ('M','F')),
CONSTRAINT di_table_Staff_email CHECK ((email LIKE '%@%') AND (email LIKE '%.%')),
CONSTRAINT di_table_Staff_sizeShirt CHECK (sizeShirt IN ('XXS','XS','S','M','L','XL','XXL','3XL')));
```


ON DELETE CASCADE:

```
Create table Vendor
(vendorID       integer          NOT NULL,
vendorName      TEXT             NOT NULL,
phoneNumber     integer          NOT NULL,
email           TEXT             NOT NULL,
address         TEXT,
TaxNumber       integer,
accountNumber   integer,
type            TEXT,
productName     TEXT,
price           integer,
unit            TEXT,
CONSTRAINT VendorPK PRIMARY KEY (vendorID),
CONSTRAINT VendorFK FOREIGN KEY (vendorID) REFERENCES Contract
        ON DELETE CASCADE
        ON UPDATE CASCADE,
CONSTRAINT di_table_Vendor_email CHECK ((email LIKE '%@%') AND (email LIKE '%.%')));
```


CREATE VIEW:

- View all information of customers who register for HGDM2024

```
CREATE VIEW HGDM24_Customerdetails AS
SELECT c.customerName, c.phoneNumber, c.email, c.gender, re.transactionID, r.raceName
FROM Customer c, Register re, Race r
WHERE re.raceID = r.raceID
AND r.raceID = 'HGDM24'
AND re.customerID = c.customerID
ORDER BY customerName;
```


THANK YOU!!