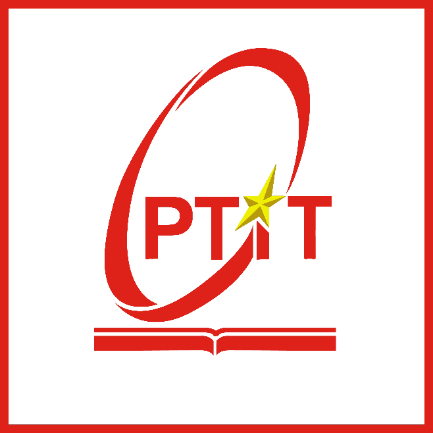
**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**Khoa Công nghệ thông tin**



**BÁO CÁO BÀI TẬP LỚN**

**MÔN: CƠ SỞ DỮ LIỆU**

**MÃ HỌC PHẦN: INT1313**

Nhóm: 8

Đề tài: Quản lý khách hàng của tiệm internet công cộng

Thành viên: - Đồng Hoàng Minh – B21DCCN522

* Lê Đình Phúc – B21DCCN593
* Hoàng Quốc Hòa – B21DCCN377
* Nguyễn Hoàng Nam – B21DCCN551
* Nguyễn Quang Minh – B21DCCN089
* Phạm Hải Dương – B21DCAT072
* Nguyễn Minh Nghĩa – B21DCAT144
* Hồ Trọng Nhật Minh – B21DCCN524

**BÁO CÁO TỔNG THỂ**

**XÂY DỰNG CƠ SỞ DỮ LIỆU QUẢN LÝ TÀI KHOẢN KHÁCH HÀNG CỦA TIỆM INTERNET CÔNG CỘNG**

**Phần 1: Mô tả kịch bản thế giới thực**

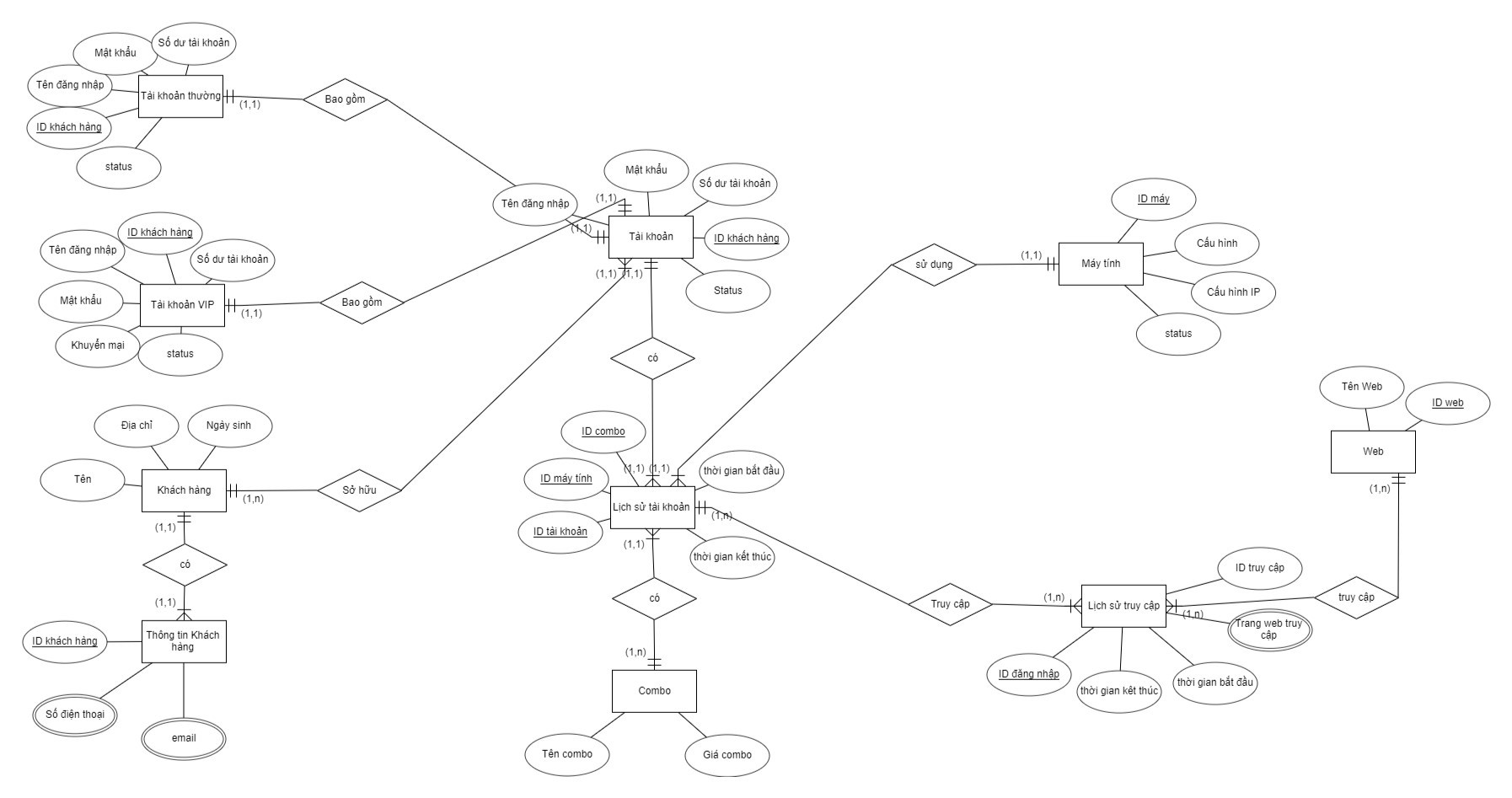
1. **Ứng dụng của hệ cơ sở dữ liệu:**

* Xây dựng hệ cơ sở dữ liệu quản lý tiệm Internet công cộng, bao gồm:
* Quản lý thông tin tiệm, thông tin về các máy tính, các dịch vụ kèm theo trong cửa hàng.
* Quản lý danh sách các khách hàng.
* Quản lý tài khoản của khách hàng, hoá đơn theo tài khoản, lịch sử truy cập của tài khoản

1. **Yêu cầu về dữ liệu:**

* Dữ liệu về khách hàng (Customer) và thông tin liên lạc của khách hàng (Contact information), bao gồm: mã khách hàng, họ tên, ngày sinh, địa chỉ, email, số điện thoại.
* Một khách hành có thể có nhiều thông tin liên lạc (nhiều email, số điện thoại)
* Dữ liệu về máy tính (Computer): thông tin của máy tính gồm mã máy, cấu hình IP của máy, tình trạng (sử dụng/ chưa sử dụng).
* Dữ liệu về tài khoản (Account): Thông tin tài khoản bao gồm mã tài khoản, tên tài khoản, mật khẩu, mã khách hàng đăng kí tài khoản, tình trạng sử dụng.
* có tài khoản thường (nomal account) và tài khoản vip (Vip account) (có thêm thuộc tính discount)
* Dữ liệu về Combo (Combo): gồm mã Combo, tên Combo và giá.
* Dữ liệu về lịch sử truy cập (Account computer history): Thông tin lịch sử truy cập bao gồm mã khách hàng, mã máy tính truy cập, loại Combo, thời gian truy cập, giá tiền
* Lịch sử truy cập web (Web access history): gồm tên web và thời gian truy cập web.
* **Điều kiện**:
* Mỗi khách hàng phải có một mã khách hàng là duy nhất.
* Mỗi máy tính phải có một mã máy tính là duy nhất.
* Mỗi tài khoản có một tên tài khoản là duy nhất.
* **Mỗi quan hệ**:
* Mối quan hệ 1-n giữa tiệm Internet và khách hàng
* Mối quan hệ 1-n giữa khách hang và thông tin liên lạc: Mỗi khách hang có thể có nhiều thông tin liên lạc.
* Mối quan hệ 1-n giữa khách hàng và tài khoản: Mỗi khách hàng có thể có nhiều tài khoản.
* normal\_account và vip\_account kế thừa account
* Mối quan hệ 3 ngôi giữa tài khoản, máy tính và combo: thể hiện qua bảng account\_computer\_history (Tại cùng 1 thời điểm, 1 account chỉ truy cập 1 máy tính, chỉ sử dụng 1 combo)
* Mỗi tài khoản có thể truy cập nhiều máy tính nhưng không được trùng cùng 1 thời điểm
* Mỗi tài khoản có thể truy cập cùng 1 máy tính nhiều lần nhưng không được trùng thời điểm
* Mỗi máy tính có thể được truy cập bởi nhiều tài khoản nhưng không trùng thời điểm
* Mối quan hệ N-N giữa account\_computer\_history và web: thể hiện qua bảng web\_access\_history:
* 1 truy cập máy (1 account\_computer\_history) có thể truy cập nhiều web cùng 1 hay khác thời điểm
* 1 web có thể được truy cập bởi nhiều truy cập máy cùng 1 hay khác thời điểm

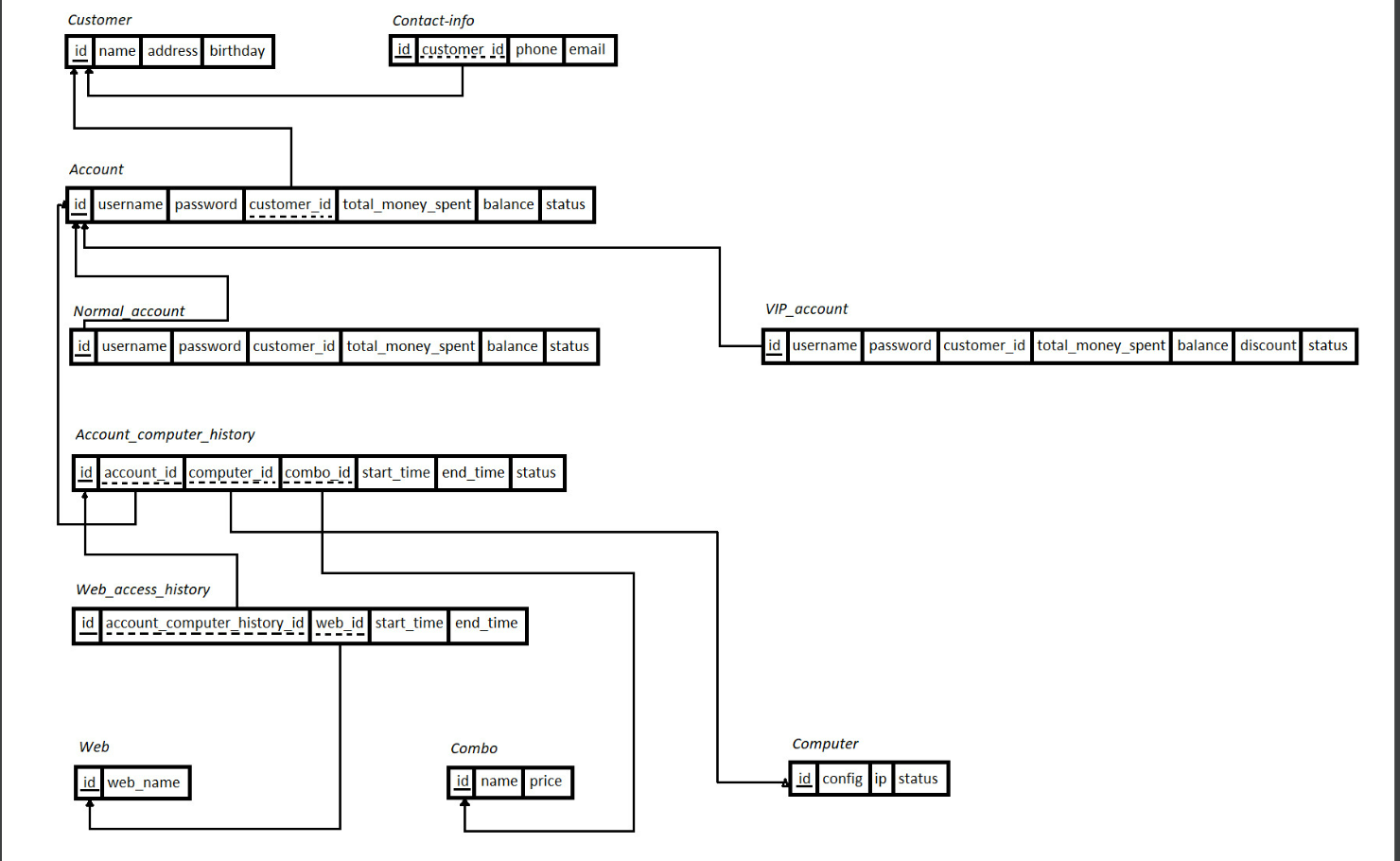
**Phần 2: Lược đồ E-R**

****

* Lược đồ gồm 10 tập thực thể, trong đó có: 2 tập thực thể con, 1 tập thực thể yếu, 1 quan hệ 3 ngôi.

**Phần 3: Chuyển lược đồ E-R sang các lược đồ quan hệ**

* Từ lược đồ E-R gồm 10 tập thực thể thành 10 lược đồ quan hệ

****

**Phần 4: Chuẩn hóa lược đồ về chuẩn 3NF**

* **Customer (id, name, address, birthday, email)**

F1 = { id->name, id->address, id->birthday }

-Lược đồ quan hệ chỉ chứa các thuộc tính nguyên tố: ở dạng chuẩn 1NF

-id là khoá dự bị, thuộc tính name, address, birthday phụ thuộc hàm đầy đủ vào id:

ở dạng chuẩn 2NF

-Do id->name, id->address, id->birthday, id là siêu khoá: ở dạng chuẩn 3NF

* **Contact\_info (id, customer\_id, phone, email)**

F2 = {id->customer-id, id->phone, id->email}

-Lược đồ quan hệ chỉ chứa các thuộc tính nguyên tố: ở dạng chuẩn 1NF

-id là khoá dự bị, thuộc tính **,** customer\_id, phone, email phụ thuộc hàm đầy đủ vào id:

ở dạng chuẩn 2NF

- Do id->customer\_id, id->phone, id->email, id là siêu khoá: ở dạng chuẩn 3NF

* **Account (id, username, password, customer\_id, total\_money\_spend, status, balance)**

F3 = {id->username, id->password, id->customer\_id, id->total\_money\_spend, id->status, id->balance}

-Lược đồ quan hệ chỉ chứa các thuộc tính nguyên tố: ở dạng chuẩn 1NF

-id là khoá dự bị, thuộc tính username, password, customer\_id, total\_money\_spend, status, balance phụ thuộc hàm đầy đủ vào id: ở dạng chuẩn 2NF

-Do id->username, id->password, id->customer\_id, id->total\_money\_spend, id->status, id->balance, id là siêu khoá: ở dạng chuẩn 3NF

* **Normal\_account (id, username, password, customer\_id, total\_money\_spend, status, balance)**

F4 = { id->username, id->password, id->customer\_id, id->total\_money\_spend, id->status, id->balance}

-Lược đồ quan hệ chỉ chứa các thuộc tính nguyên tố: ở dạng chuẩn 1NF

-id là khoá dự bị, thuộc tính password, customer\_id, total\_money\_spend, status phụ thuộc hàm đầy đủ vào id: ở dạng chuẩn 2NF

- Do id->username, id->password, id->customer\_id, id->total\_money\_spend, id->status, id->balance, id là siêu khoá: ở dạng chuẩn 3NF

* **VIP\_account (id, username, password, customer\_id, total\_money\_spend, discount, status, balance)**

F5 = { id->password, id->password, id->customer\_id, id->total\_money\_spend, id->discount, id->status, id->balance }

-Lược đồ quan hệ chỉ chứa các thuộc tính nguyên tố: ở dạng chuẩn 1NF

-id là khoá dự bị, thuộc tính username, password, customer\_id, total\_money\_spend, discount, status, balance phụ thuộc hàm đầy đủ vào id: ở dạng chuẩn 2NF

-Do id->username, id->password, id->customer\_id, id->total\_money\_spend, id->discount, id->status, id->balance, id là siêu khoá: ở dạng chuẩn 3NF

* **Account\_computer\_history (id, account\_id, computer\_id, combo\_id, start\_time, end\_time, status)**

F6 = {id-> account\_id, id-> computer\_id, id-> combo\_id, id-> start\_time , id-> end\_time, id->status}

-Lược đồ quan hệ chỉ chứa các thuộc tính nguyên tố: ở dạng chuẩn 1NF

-id là khoá dự bị, thuộc tính account\_id, computer\_id, combo\_id, start\_time, end\_time, status phụ thuộc hàm đầy đủ vào id: ở dạng chuẩn 2NF

-Do id-> account-id, id-> computer-id, id-> combo-id, id-> start\_time , id-> end\_time, id là siêu khoá: ở dạng chuẩn 3NF

* **Web\_access\_history (id, account\_computer\_id, web\_id, start\_time, end\_time)**

F7 = {id->account\_computer\_id, id->web\_id, id->start\_time, id->end\_time}

-Lược đồ quan hệ chỉ chứa các thuộc tính nguyên tố: ở dạng chuẩn 1NF

-id là khoá dự bị, thuộc tính account\_computer\_id, web\_id, start\_time, end\_time phụ thuộc hàm đầy đủ vào id: ở dạng chuẩn 2NF

-Do id->account\_computer\_id, id->web\_id, id->start\_time, id->end\_time, id là siêu khoá: ở dạng chuẩn 3NF

* **Web (id, web\_name)**

F8 = {id->web\_name }

-Lược đồ quan hệ chỉ chứa các thuộc tính nguyên tố: ở dạng chuẩn 1NF

-id là khoá dự bị, thuộc tính web\_name phụ thuộc hàm đầy đủ vào id: ở dạng chuẩn 2NF

-Do id->web\_name, id là siêu khoá: ở dạng chuẩn 3NF

* **Combo (id, name, price)**

F9 = {id->name, id->price}

-Lược đồ quan hệ chỉ chứa các thuộc tính nguyên tố: ở dạng chuẩn 1NF

-id là khoá dự bị, thuộc tính name, price phụ thuộc hàm đầy đủ vào id: ở dạng chuẩn 2NF

-Do id->name, id->price, id là siêu khoá: ở dạng chuẩn 3NF

* **Computer (id, config, ip, status)**

F10 = {id->name, id->config, id->ip, id->status}

-Lược đồ quan hệ chỉ chứa các thuộc tính nguyên tố: ở dạng chuẩn 1NF

-id là khoá dự bị, thuộc tính config, ip, status phụ thuộc hàm đầy đủ vào id:

ở dạng chuẩn 2NF

-Do id->name, id->config, id->ip, id->status, id là siêu khoá: ở dạng chuẩn 3NF

* **Sau khi chuẩn hóa 10 lược đồ, được 10 lược đồ ở dạng chuẩn 3NF**.

**Phần 5: Cài đặt hệ thống**

* Cài đặt các bảng của csdl và thêm dữ liệu vào các bảng

CREATE DATABASE quanlytiemnet

go

USE quanlytiemnet

CREATE TABLE customer (

Customer\_id INT NOT NULL PRIMARY KEY,

name NVARCHAR(45) NULL,

address NVARCHAR(45) NULL,

birthday DATE NULL

);

go

CREATE TABLE contact\_info (

Contact\_id INT NOT NULL PRIMARY KEY,

Customer\_id INT NULL,

phone NVARCHAR(45) NULL,

email NVARCHAR(45) NULL,

FOREIGN KEY (Customer\_id) REFERENCES customer(Customer\_id)

);

go

CREATE TABLE account (

Account\_id INT NOT NULL PRIMARY KEY,

username NVARCHAR(45) NULL,

password NVARCHAR(45) NULL,

customer\_id INT NOT NULL,

balance BIGINT NULL,

total\_money\_spent BIGINT NULL,

status NVARCHAR(45) NULL,

FOREIGN KEY (customer\_id) REFERENCES customer(Customer\_id)

);

go

CREATE TABLE normal\_account (

Account\_id INT NOT NULL UNIQUE,

username NVARCHAR(45) NULL,

password NVARCHAR(45) NULL,

customer\_id INT NULL,

balance BIGINT NULL,

total\_money\_spent BIGINT NULL,

status NVARCHAR(45) NULL,

PRIMARY KEY(Account\_id),

FOREIGN KEY (Account\_id) REFERENCES account(Account\_id)

);

go

CREATE TABLE vip\_account (

Account\_id INT NOT NULL UNIQUE,

username NVARCHAR(45) NULL,

password NVARCHAR(45) NULL,

customer\_id NVARCHAR(45) NULL,

balance BIGINT NULL,

total\_money\_spent BIGINT NULL,

discount NVARCHAR(45) NULL,

status NVARCHAR(45) NULL,

PRIMARY KEY(Account\_id),

FOREIGN KEY (Account\_id) REFERENCES account(Account\_id)

);

go

CREATE TABLE combo (

Combo\_id INT NOT NULL PRIMARY KEY,

name NVARCHAR(45) NULL,

price INT NULL

);

go

CREATE TABLE computer (

Computer\_id INT NOT NULL PRIMARY KEY,

config NVARCHAR(255) NULL,

ip NVARCHAR(45) NULL,

status NVARCHAR(45) NULL

);

go

CREATE TABLE account\_computer\_history (

Access\_id INT NOT NULL PRIMARY KEY,

Account\_id INT NOT NULL,

Computer\_id INT NOT NULL,

Combo\_id INT NOT NULL,

start\_time DATETIME NULL,

end\_time DATETIME NULL,

status NVARCHAR(45) NULL,

FOREIGN KEY (Account\_id) REFERENCES account(Account\_id),

FOREIGN KEY (Combo\_id) REFERENCES combo(Combo\_id),

FOREIGN KEY (Computer\_id) REFERENCES computer(Computer\_id)

);

go

CREATE TABLE web (

Web\_id INT NOT NULL PRIMARY KEY,

Web\_name NVARCHAR(45) NULL

);

go

CREATE TABLE web\_access\_history (

WebAccess\_id INT NOT NULL PRIMARY KEY,

Access\_id INT NULL,

Web\_id INT NULL,

start\_time DATETIME NULL,

end\_time DATETIME NULL,

FOREIGN KEY (Access\_id) REFERENCES account\_computer\_history(Access\_id),

FOREIGN KEY (Web\_id) REFERENCES web(Web\_id)

);

go

--Add data to table

INSERT INTO customer (Customer\_id, name, address, birthday)

VALUES

(1, 'Phúc', 'Hà Đông', '2002-02-12'),

(2, 'Nam', 'Cầu Diễn', '2001-05-24'),

(3, 'Dương', 'Cầu Giấy', '2004-11-21'),

(4, 'Nghĩa', 'Thanh Xuân', '2003-09-04');

go

--

INSERT INTO contact\_info (Contact\_id, Customer\_id, phone, email)

VALUES

(1, 1, '0904738183', 'phuc@gmail.com'),

(2, 2, '0372847182', 'nam@gmail.com'),

(3, 2, '0946838223', 'nam@gmail.com'),

(4, 3, '0988482123', 'duong@gmail.com'),

(5, 4, '0372884999', 'nghia@gmail.com');

go

--

-- Chèn dữ liệu vào bảng account

INSERT INTO account (Account\_id, username, password, customer\_id, balance, total\_money\_spent, status)

VALUES (1, 'phuc3', '3827', 1, 12000, 387000, 'Online'),

(2, 'phuc1', '4821', 1, 20000, 102040, 'Offline'),

(3, 'namz', '9482', 2, 50000, 1040300, 'Offline'),

(4, 'duong34', '8291', 3, 5000, 14900, 'Online'),

(5, 'nghia96', '0492', 4, 23000, 69300, 'Offline');

go

INSERT INTO normal\_account (Account\_id, username, password, customer\_id, balance, total\_money\_spent, status)

VALUES

(2, 'phuc1', '4821', 1, 20000, 102040, 'Offline'),

(4, 'duong34', '8291', 3, 5000, 14900, 'Online'),

(5, 'nghia96', '0492', 4, 23000, 69300, 'Offline');

go

--

INSERT INTO vip\_account (Account\_id, username, password, customer\_id, balance, total\_money\_spent, discount, status)

VALUES

(1, 'phuc3', '3827', '1', 12000, 387000, '30%', 'Online'),

(3, 'namz', '9482', '2', 50000, 1040300, '30%', 'Offline');

go

--

INSERT INTO account\_computer\_history (Access\_id, Account\_id, Computer\_id, Combo\_id, start\_time, end\_time, status)

VALUES

(1, 4, 2, 2, '2023-09-17 15:10:00', '2023-09-17 16:10:00', 'complete'),

(2, 1, 5, 1, '2023-09-17 23:00:00', '2023-09-18 07:00:00', 'complete'),

(3, 3, 1, 1, '2023-09-18 23:00:00', '2023-09-19 07:00:00', 'complete'),

(4, 5, 3, 1, '2023-09-19 06:00:00', '2023-09-19 12:00:00', 'compelte'),

(5, 3, 1, 2, '2023-09-20 06:00:00', '2023-09-20 12:00:00', 'complete'),

(6, 2, 4, 3, '2023-09-20 07:00:00', '2023-09-20 08:00:00', 'complete'),

(7, 3, 2, 1, '2023-09-20 23:00:00', '2023-09-21 07:00:00', 'complete'),

(8, 1, 3, 2, '2023-09-21 12:00:00', '2023-09-21 19:00:00', 'pending'),

(9, 4, 1, 2, '2023-09-21 14:00:00', '2023-09-21 18:00:00', 'pending');

go

--

INSERT INTO combo (Combo\_id, name, price)

VALUES

(1, '23h-7h', 24000),

(2, 'tính theo giờ', 10000),

(3, '6h-12h', 30000);

go

--

INSERT INTO computer (Computer\_id, config, ip, status)

VALUES

(1, 'Core I5 12400F|Ram 16G| GTX 1660 Super 6G| SSD 240G', '192.168.2.3', 'Active'),

(2, 'Core I3 9100F| Ram 8G| RX 580 8G| SSD 256G', '192.168.8.4', 'Inactive'),

(3, 'Core I5 10400F| GTX 1660 Super 6G| Ram 8G| SSD 240G', '192.168.2.1', 'Active'),

(4, 'Dual Xeon E5 2676 V3 | RAM 64G | RX 580 8G | NVME 500G', '192.168.9.0', 'Inactive'),

(5, 'Xeon E5 2676 V3 | RAM 64G | GT 1030 4G | NVME 500G', '192.168.0.1', 'Inactive');

go

--

INSERT INTO web (Web\_id, Web\_name)

VALUES

(1, 'facebook.com'),

(2, 'google.com'),

(3, 'youtube.com'),

(4, 'code.ptit.edu.vn'),

(5, 'qldt.ptit.edu.vn'),

(6, 'github.com');

go

--

INSERT INTO web\_access\_history (WebAccess\_id, Access\_id, Web\_id, start\_time, end\_time)

VALUES

(1, 1, 1, '2023-09-17 15:20:00', '2023-09-17 15:30:00'),

(2, 2, 6, '2023-09-17 23:01:00', '2023-09-18 3:08:00'),

(3, 2, 3, '2023-09-17 23:01:00', '2023-09-18 1:01:00'),

(4, 4, 2, '2023-09-19 06:05:00', '2023-09-19 07:00:00'),

(5, 7, 4, '2023-09-20 23:01:00', '2023-09-21 01:00:00'),

(6, 3, 1, '2023-09-20 07:20:00', '2023-09-20 11:30:00'),

(7, 5, 6, '2023-09-20 06:01:00', '2023-09-20 09:08:00'),

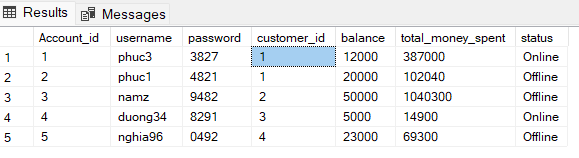
(8, 6, 3, '2023-09-20 07:00:00', '2023-09-20 08:00:00'),

(9, 5, 2, '2023-09-20 09:30:00', '2023-09-19 10:00:00'),

(10, 8, 2, '2023-09-21 12:05:00', '2023-09-21 17:00:00'),

(11, 9, 4, '2023-09-21 15:01:00', '2023-09-21 17:00:00');

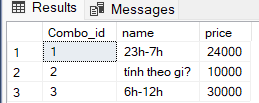
1. **Bảng Account: thông tin về tài khoản của khách hàng**

****

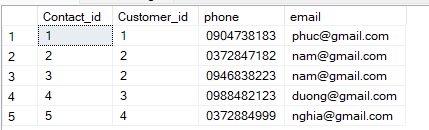
1. **Bảng Computer: thông tin về máy tính**

****

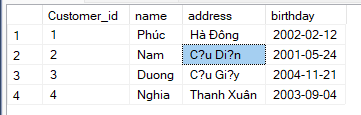
1. **Bảng Combo: thông tin về các combo của quán**

****

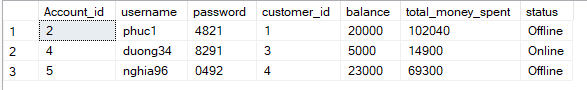
1. **Bảng Contact\_info: Thông tin liên lạc của khách hàng**

****

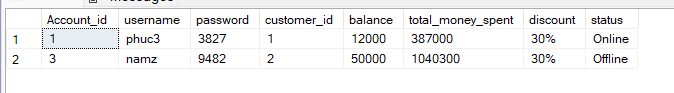
1. **Bảng Customer: Thông tin của khách hang**

****

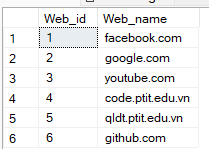
1. **Bảng normal\_account: Thông tin về tài khoản loại thường**

****

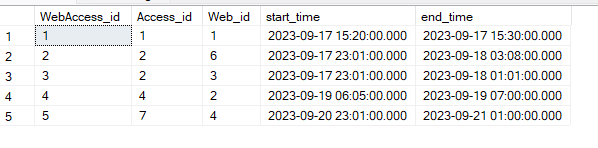
1. **Bảng vip\_account: Thông tin về tài khoản loại vip**

****

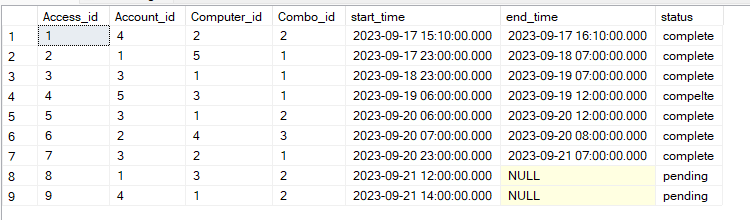
1. **Bảng web: thông tin về các website**

****

1. **Bảng web\_access\_history: Thông tin về website mà khách hàng truy cập vào trong thời gian sử dụng**

****

1. **Bảng account\_computer\_history: Thông tin về thời gian sử dụng máy ở tiệm internet của khách hàng**

****

* **Các câu truy vấn dữ liệu**

1. Tên web, lượng truy cập trong ngày 2023-9-17

select web\_name,count from(

(select web\_id,count(WebAccess\_id) as count from web\_access\_history where YEAR(start\_time) = 2023 AND MONTH(start\_time) = 9 AND DAY(start\_time) = 17 GROUP BY web\_id) tmp

left join web on tmp.Web\_id = web.Web\_id

)

1. tên combo, lượng sử dụng với các tài khoản có total\_money\_spent >= 100000

select name,count from(

(select combo\_id,count(Account\_id) as count from account\_computer\_history where account\_id in(

select Account\_id from account where total\_money\_spent >= 100000

) group by Combo\_id ) tmp

left join combo on tmp.combo\_id = combo.Combo\_id

)

1. Tên khách hàng, các trang web từng truy cập

SELECT tmp3.name, web.web\_name FROM (

(SELECT customer.name,tmp2.web\_id FROM (

(SELECT account.customer\_id, tmp.web\_id FROM (

(SELECT DISTINCT account\_id, web\_id FROM

account\_computer\_history LEFT JOIN web\_access\_history ON account\_computer\_history.Access\_id=web\_access\_history.Access\_id

WHERE web\_id IS NOT NULL) tmp

LEFT JOIN account ON tmp.account\_id=account.Account\_id

)) tmp2

LEFT JOIN customer ON tmp2.customer\_id=customer.Customer\_id

)) tmp3

LEFT JOIN web ON tmp3.web\_id=web.Web\_id

) ORDER BY name ASC