**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA HỆ THỐNG THÔNG TIN**

--------------------------------------------------------------



**ĐỒ ÁN MÔN HỌC HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

**ĐỀ TÀI: QUẢN LÝ QUÁN CAFE**

-------------------------------------------------------------------------------

Giảng viên hướng dẫn: ***ThS. ĐỖ THỊ MINH PHỤNG***

Sinh viên thực hiện: ***NGUYỄN KIM HOÀNG* 16520438**

***PHẠM LAM TRƯỜNG*  17521188**

***CAO HOÀNG TÚ* 17521192**

***TRẦN VĂN ĐIỆP* 18520601**

Lớp : **IS210.K21**

**NỘI DUNG**

[Chương 1: TỔNG QUAN ĐỀ TÀI 4](#_Toc44074302)

[1.1 Đặt vấn đề 4](#_Toc44074303)

[1.2 Giải pháp 4](#_Toc44074304)

[1.3 Mục tiêu 5](#_Toc44074305)

[1.4 Công cụ sử dụng 5](#_Toc44074306)

[Chương 2: PHÂN TÍCH YÊU CẦU VÀ THIẾT KẾ DỮ LIỆU 5](#_Toc44074307)

[2.1 Phân tích yêu cầu 5](#_Toc44074308)

[2.1.1 Yêu cầu chức năng 5](#_Toc44074309)

[2.1.1.1 Yêu cầu lưu trữ 5](#_Toc44074310)

[2.1.1.2 Yêu cầu đáp ứng hệ thống 7](#_Toc44074316)

[2.1.1.3 Yêu cầu về tính năng 7](#_Toc44074317)

[2.1.1.4 Yêu cầu từ phía người dùng 7](#_Toc44074318)

[2.1.2Yêu cầu phi chức năng 8](#_Toc44074319)

[2.2 Thiết kế mô hình quan hệ 8](#_Toc44074320)

[2.2.1Mô hình dữ liệu quan hệ 8](#_Toc44074321)

[2.2.2Bảng mô tả chi tiết bảng 9](#_Toc44074327)

[2.2.2.1 Table: NHANVIEN 9](#_Toc44074328)

[2.2.2.2 Table: BAN 9](#_Toc44074335)

[2.2.2.3 Table: LOAIDOUONG 10](#_Toc44074336)

[2.2.2.4 Table: DOUONG 10](#_Toc44074345)

[2.2.2.5 Table: ORDERS 10](#_Toc44074346)

[2.2.2.6 Table: CTORDERS 11](#_Toc44074347)

[2.2.2.7 Table: NGUYENVATLIEU 11](#_Toc44074348)

[2.2.2.8 Table: PHIEUNHAPHANG 11](#_Toc44074349)

[2.2.2.9 Table: CTPHIEUNHAP 12](#_Toc44074350)

[2.2.2.10 Table: PHIEUXUATHANG 12](#_Toc44074351)

[2.2.2.11 Table: CTPHIEUXUAT 12](#_Toc44074352)

[2.2.2.12 Table: KHUYENMAI 13](#_Toc44074353)

[2.3 Ràng buộc toàn vẹn dữ liệu 13](#_Toc44074354)

[2.3.1Trạng thái của bàn chỉ được nhận 3 giá trị (Trống, Đang phục vụ, Đang chờ) 13](#_Toc44074355)

[2.3.2Thành tiền của CTHD phải bằng SOLUONG\*GIABAN 13](#_Toc44074362)

[2.3.3Trị giá hóa đơn bằng tổng thành tiền của CTHD ứng với mỗi mã hóa đơn 13](#_Toc44074363)

[2.3.4Ngày kết thúc khuyến mãi phải lớn hơn hoặc bằng ngày bắt đầu khuyến mãi 14](#_Toc44074364)

[2.3.5Thành tiền của CTPHIEUNHAP phải bằng SOLUONG\*DONGIA 14](#_Toc44074365)

[2.3.6Trị giá của phiếu nhập hàng phải bằng tổng THANHTIEN ứng với mỗi MAPHIEUNHAP 14](#_Toc44074366)

[Chương 3: KHÁI QUÁT MÔ HÌNH VÀ CÔNG NGHỆ SỬ DỤNG 15](#_Toc44074367)

[3.1 Khái quát về hệ quản trị cơ sở dữ liệu Oracle 15](#_Toc44074369)

[3.1.1Tổng quan về cơ sở dữ liệu Oracle 15](#_Toc44074370)

[3.1.2Tổng quan về kiến trúc Oracle 15](#_Toc44074371)

[3.2 Khái quát về mô hình MVC 16](#_Toc44074372)

[3.2.1MVC là gì? 16](#_Toc44074373)

[3.2.2Các thành phần trong MVC 16](#_Toc44074374)

[3.2.3Quá trình hoạt động của mô hình MVC 17](#_Toc44074375)

[3.2.4Ưu và nhược điểm của mô hình MVC 18](#_Toc44074376)

[2.4.4.1 Ưu điểm 18](#_Toc44074383)

[2.4.4.2 Nhược điểm 18](#_Toc44074384)

[3.3 Khái quát về JDBC 18](#_Toc44074385)

[3.3.1JDBC là gì? 18](#_Toc44074386)

[3.3.2Thành phần của JDBC 19](#_Toc44074387)

[3.4 Khái quát về JavaSwing 19](#_Toc44074388)

[3.5 Khái quát về Netbeans 8.2 19](#_Toc44074389)

[3.5.1Giới thiệu phần mềm NetBeans. 19](#_Toc44074390)

[3.5.2Ưu điểm phần mềm NetBeans. 20](#_Toc44074391)

[3.5.3Các tính năng của phần mềm NetBeans. 21](#_Toc44074392)

[3.6 Khái quát về Git 21](#_Toc44074393)

[3.7 Khái quát về GitKraken 21](#_Toc44074394)

[Chương 4: XÂY DỰNG VÀ QUẢN LÝ GIAO TÁC 22](#_Toc44074395)

[4.1 Function 22](#_Toc44074397)

[4.2 Stored Procedure 22](#_Toc44074398)

[4.3 Trigger 22](#_Toc44074399)

[Chương 5: XỬ LÝ TRUY XUẤT ĐỒNG THỜI 22](#_Toc44074400)

[5.1 Các mức cô lập trong Oracle 22](#_Toc44074402)

[5.2 Lost Update 22](#_Toc44074403)

[5.3 Non - Repeatable Read 26](#_Toc44074404)

[5.4 Phantom Read 30](#_Toc44074405)

[5.5 DeadLock 30](#_Toc44074406)

# **Chương 1:** **TỔNG QUAN ĐỀ TÀI**

## Đặt vấn đề

Quán cafe Glass Coffee là một quán cafe nằm trong nội thành thành phố. Với lượng khách hàng ngày càng tăng, để phục vụ khách hàng được nhanh chóng, tốt hơn, mặt khác giúp cho việc quản lý quán cafe được tốt hơn nên chủ quán đang từng bước tin học hoá các khâu quản lý. Đặc biệt là trong công tác quản lý nguyên vật liệu, quản lý bán hàng, quản lý nhân viên, quản lý các thông tin khuyến mãi. Bởi vì công tác thủ công trước đó của quán gặp phải những vấn đề như sau:

* Tra cứu thông tin về hàng hoá, các đại lý cung cấp cũng như hoá đơn nhập hàng mất nhiều thời gian, đôi khi bị mất mát giấy tờ.
* Thông tin về các loại hàng hoá, hoá đơn ngày một nhiều khiên cho việc lưu trữ bằng giấy tờ trở nên cồng kềnh và khó quản lý.
* Việc báo cáo thống kê thu chi phải thực hiện thủ công rất mất thời gian và đôi khi xảy ra sai sót.
* Khi khách hàng vào quán đông thì thu ngân gặp rất nhiều khó khăn khi vừa phải thanh toán tiền cho bàn này vừa phải nhận order mới của bàn khác dẫn đến không thể phục vụ khách hàng một cách nhanh chóng.

Trước tình hình đó, với vai trò là một nhà phát triển phần mềm quản lý, được quán Glass coffee yêu cầu xây dựng một hệ thống thông tin để khắc phục được các vấn đề như:

* Giảm khối lượng lưu trữ bằng giấy.
* Cập nhật dữ liệu nhanh chóng, chính xác.
* Thống kê số lượng hàng hoá, thu chi và tình hình doanh thu của quán.
* Lưu trữ được lượng dữ liệu lớn, lâu dài và an toàn.
* Hỗ trợ việc quản lý order và thanh toán tiền cho khách hàng nhanh chóng, hiệu quả.

Do trước đây quán cafe chưa được tin học hoá trong khâu quản lý nên đây là một hệ thống hoàn toàn mới. Hệ thống cho phép cả người quản lý và thu ngân đăng nhập vào sử dụng. Với mỗi nhóm người dùng sẽ được sử dụng một số chức năng của hệ thống riêng quản lý của quán có thể sử dụng tất cả. Cụ thể, hệ thống cần có các chức năng quản lý thức uống, quản lý bàn, quản lý nguyên vật liệu, quản lý nhân viên và báo cáo thống kê. Thêm vào đó còn có chức năng quản lý thông tin khuyến mãi.

## Giải pháp

Theo như tình trạng hiện tại và nhu cầu quản lý quán cafe của quản lý quán và chủ quán, giải pháp được nhóm chúng em đề xuất là xây dựng nên một hệ thống quản lý quán cafe mới thay cho hệ thống quản lý bằng tay cũ để công việc quản lý thông tin của “Glass Coffee” trở nên dễ dàng và chính xác. Đồng thời, hệ thống giúp nâng cao tính quản lý chặt chẽ thu chi của quán, nâng cao năng suất lao động của nhân viên, tiết kiệm chi phí hoạt động, tìm kiếm, lưu trữ và báo cáo thống kê thông tin một cách dễ dàng. Bên cạnh đó các quy trình nghiệp vụ với sự hỗ trợ đắc lực của phần mềm cũng sẽ được xử lý nhanh chóng và hiệu quả.

## Mục tiêu

* Xây dựng ứng dụng đáp ứng những nhu cầu dưới đây:
* Đáp ứng được nhu cầu quản lý các thông tin: nhân viên, khách hàng, xuất nhập hàng, doanh thu, báo cáo thống kê,… Một cách hợp lí và chính xác nhất.
* Giảm nhân lực thủ công, giảm chi phí cho việc ghi chép bằng tay như truyền thống.
* Hệ thống hóa được quy trình nghiệp vụ của thư viện.
* Tiết kiệm thời gian, công sức, và giảm tối thiểu thời gian hỗ trợ khách hàng trong lúc bán hàng.
* Cài đặt các trigger, function, stored procedure, transaction.
* Thiết kế giao diện giúp người dùng dễ dàng sử dụng hệ thống thông qua các nút chọn.
* Sử dụng các mức cô lập để giải quyết vấn đề xử lý đồng thời.

## Công cụ sử dụng

* Trong quá trình thực hiện đồ án xây dựng hệ thông quản lý quán cafe Glass Coffee nhóm đã sử dụng một số phần mềm hỗ trợ sau đây:
* Oracle developer
* Netbean 8.2
* Git
* GitKraken
* Ireport

# **Chương 2: PHÂN TÍCH YÊU CẦU VÀ THIẾT KẾ DỮ LIỆU**

## Phân tích yêu cầu

### Yêu cầu chức năng

#### Yêu cầu lưu trữ

Yêu cầu quan trọng nhất của hệ thống là tính năng lưu trữ cơ sở dữ liệu của quán cafe qua các năm. Vì đó là nền tảng để xây những chức năng của hệ thống. Vậy nên chúng ta cần phải quản lý và lưu trữ những thông tin sau:

|  |  |
| --- | --- |
| **Đối tượng cần quản lý** | **Dữ liệu nào cần lưu trữ** |
| Thông đăng nhập | * (ID)Tên đăng nhập. * (Password) mật khẩu. * Loại nhân viên. |
| Thông tin nhân viên | * Mã nhân viên. * Họ tên nhân viên. * Ngày sinh. * CMND. * Địa chỉ. * Số điện thoại. * Ngày vào làm. * Loại nhân viên. |
| Thông tin đồ uống | * Mã đồ uống. * Tên đồ uống. * Giá bán. * Loại đồ uống. |
| Thông tin nguyên vật liệu | * Mã NVL. * Tên NVL. * Giá mua. * Ngày sản xuất. * Số lượng. * Đơn vị tính. |
| Thông tin đặt món | * Mã đặt món. * Danh sách đồ uống. * Danh sách đồ uống tặng kèm (nếu có). |
| Thông tin bàn | * Mã bàn. * Tên bàn. * Tình trạng. |
| Thông tin phiếu nhập hàng | * Mã phiếu nhập. * Ngày lập phiếu. * Danh sách hàng nhập. * Trị giá. * Ghi chú. |
| Thông tin hóa đơn | * Mã số hóa đơn. * Ngày lập hóa đơn. * Danh sách đồ uống. * Thành tiền. |
| Thông tin khuyến mãi | * Mã khuyến mãi. * Tên khuyến mãi. * Ngày bắt đầu. * Ngày kết thúc. * Ghi chú. |
| Thông tin phiếu xuất hàng | * Mã phiếu xuất hàng. * Ngày lập. * Trị giá. |



#### Yêu cầu đáp ứng hệ thống

* + - Hệ thống xây dựng lên phải đáp ứng được nhu cầu quản lý nhân viên, NVL và thông tin liên quan đến doanh thu của quán.
    - Cập nhật thông tin chính xác, nhanh chóng để đẩy mạnh tiến độ công việc của quán.
    - Hệ thống cần phải được bảo mật kĩ càng các thông tin của đối tượng được lưu trữ.
    - Hệ thống hỗ trợ thống kê, báo cáo doanh thu theo yêu cầu của chủ quán.
    - Dễ dàng nâng cấp và tăng tính năng tiện ích khác.
    - Hệ thống phù hợp với những loại máy móc sẵn có của quán.

#### Yêu cầu về tính năng

* Chức năng tính tiền hóa đơn: Tự động tính tiền cho hoá đơn thanh toán của khách hàng. Khi nhân viên thu ngân có yêu cầu thanh toán thì hệ thống tự động căn cứ vào order của bàn và thông tin giá bán trong cơ sở dữ liệu để thực hiện tính tiền cho hoá đơn. Nếu có khuyến mãi giảm giá thì tự động trừ tiền ra.
* Chức năng thống kê, báo cáo: đáp ứng nhu cầu của chủ quán để xem xét doanh thu và có những đúc kết, hướng đi của quán.
* Chức năng tra cứu: đáp ứng được nhu cầu của người dùng nhanh chóng như: tra cứu thông tin đồ uống, tra cứu thông tin hóa đơn, tra cứu thông tin hóa đơn, …v.v….
* Chức năng cập nhật thông tin: thêm, cập nhật, xóa thông tin của các đối tượng lưu trữ: Nhân viên, bàn, NVL, khuyến mãi, hóa dơn, phiếu nhập hàng, đồ uống.
* Chức năng phân quyền truy cập.

#### Yêu cầu từ phía người dùng

* Giao diện đẹp thuận tiện, dễ sử dụng với mọi mọi người dùng
* Các bước đăng nhập vào hệ thống dễ dàng, trải qua ít các bước
* Thông tin hiển thị chi tiết đầy đủ, duy trì lưu trữ dữ liệu lâu dài trong hệ thống trong phạm vi thời gian nhất định.
* Hệ thống hoạt động ổn định theo thời gian
* Hoàn thành đúng thời gian theo hợp đồng đã thỏa thuận.

### Yêu cầu phi chức năng

* Yêu cầu giao diện: tiện dụng, thân thiện, sinh động và tương tác cao
* Yêu cầu hiệu suất:
* Phần mềm sử dụng hợp lý tài nguyên hệ thống để tránh treo máy, tràn bộ nhớ ảnh hưởng các chương trình khác.
* Yêu cầu an toàn:
* Phần mềm chạy trên hệ thống không xung đột với các phần mềm khác.
* Yêu cầu bảo mật:

|  |  |  |
| --- | --- | --- |
| **Chức năng** | **Nhân viên** | **Người quản lý** |
| Quản lý nhân viên |  | X |
| Quản lý NVL |  | X |
| Quản lý thông tin khuyến mãi |  | X |
| Quản lý thông tin đồ uống |  | X |
| Quản lý bán hàng | X | X |
| Quản lý bàn |  | X |
| Thống kê báo cáo | X |  |
| Đổi mật khẩu | X | X |
| Đăng nhập | X | X |

## Thiết kế mô hình quan hệ

### Mô hình dữ liệu quan hệ



### Bảng mô tả chi tiết bảng

#### Table: NHANVIEN

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Diễn giải | Kiểu dữ liệu | Chiều dài | Miền giá trị | Ghi chú |
| 1 | MANV | Mã nhân viên | Char | 5 |  | Khóa chính |
| 2 | HOTEN | Họ tên | Text | 50 |  |  |
| 3 | NGAYSINH | Ngày sinh | Date |  |  |  |
| 4 | DIACHI | Địa chỉ | Text | 100 |  |  |
| 5 | CMND | Số CMND | Text | 20 |  |  |
| 6 | SDT | Số điện thoại | Text | 20 |  |  |
| 7 | NGVL | Ngày vào làm | Date |  |  |  |
| 8 | PASSWORD | Mật khẩu | Text | 20 |  |  |
| 9 | LOAINV | Loại nhân viên | Text | 2 | NV,QL |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Diễn giải | Kiểu dữ liệu | Chiều dài | Miền giá trị | Ghi chú |
| 1 | MABAN | Mã bàn | Number | 3 |  | Khóa chính |
| 2 | TENBAN | Tên Bàn | Text | 50 |  |  |
| 3 | TRANGTHAI | Trạng thái | Text | 20 | Trống, Đang chờ, Đang phục vụ |  |



#### Table: BAN

#### Table: LOAIDOUONG

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Diễn giải | Kiểu dữ liệu | Chiều dài | Miền giá trị | Ghi chú |
| 1 | MALOAIDOUONG | Mã loại đồ uống | Char | 2 | ST, CF, NN, KH | Khóa chính |
| 2 | TENLOAIDOUONG | Tên loại đồ uống | Text | 50 |  |  |
| 3 | HINHANH | Hình ảnh minh họa | Text | 50 |  |  |



#### Table: DOUONG

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Diễn giải | Kiểu dữ liệu | Chiều dài | Miền giá trị | Ghi chú |
| 1 | MADOUONG | Mã đồ uống | Char | 5 |  | Khóa chính |
| 2 | TENDOUONG | Tên đồ uống | Text | 50 |  |  |
| 3 | GIABAN | Giá đồ uống | Number | 10 |  |  |
| 4 | MALOAIDOUONG | Mã loại đồ uống | Char | 2 | ST, CF, NN, KH | Khóa ngoại |
| 5 | HINHANH | Hình ảnh minh họa | Text | 50 |  |  |

#### Table: ORDERS

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Diễn giải | Kiểu dữ liệu | Chiều dài | Miền giá trị | Ghi chú |
| 1 | MAODERS | Mã phiếu order | Number | 3 |  | Khóa chính |
| 2 | NGAYLAP | Ngày lập phiếu | Date |  |  |  |
| 4 | MABAN | Mã bàn phục vụ | Number | 3 |  | Khóa ngoại |
| 5 | TRANGTHAI | Trạng thái | Text |  |  |  |
| 6 | TRIGIA | Trị giá hóa đơn | Number | 10 |  |  |
| 7 | MAKM | Mã khuyến mãi | Number | 3 |  |  |
| 8 | PHAITRA | Số tiền phải trả | Number | 10 |  |  |

#### Table: CTORDERS

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Diễn giải | Kiểu dữ liệu | Chiều dài | Miền giá trị | Ghi chú |
| 1 | MAORDER | Mã phiếu order | Char | 5 |  | Khóa ngoại đồng thời là khóa chính |
| 2 | MADOUONG | Mã đồ uống | Char | 5 |  | Khóa ngoại đồng thời là khóa chính |
| 3 | GIABAN | Giá bán | Number | 10 |  |  |
| 4 | SOLUONG | Số lượng | Number | 3 |  |  |
| 5 | THANHTIEN | Thành tiền | Number | 10 |  |  |

#### Table: NGUYENVATLIEU

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Diễn giải | Kiểu dữ liệu | Chiều dài | Miền giá trị | Ghi chú |
| 1 | MANVL | Mã NVL | Char | 5 |  | Khóa chính |
| 2 | TENNVL | Tên NVL | Text | 50 |  |  |
| 3 | SOLUONG | Số lượng | Number | 3 | > 0 |  |
| 4 | DONGIA | Đơn giá NVL | Number | 10 | > 0 |  |
| 5 | DONVITINH | Đơn vị tính | Text | 20 |  |  |

#### Table: PHIEUNHAPHANG

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Diễn giải | Kiểu dữ liệu | Chiều dài | Miền giá trị | Ghi chú |
| 1 | MAPHIEUNHAP | Mã phiếu nhập | Char | 5 |  | Khóa chính |
| 2 | NGAYLAP | Ngày lập phiếu | Date | 50 |  |  |
| 3 | NVLAP | Mã NV lập | Char | 5 |  | Khóa ngoại |
| 4 | TRANGTHAI | Trạng thái | Text | 20 | Chưa nhận hàng, Đã nhận hàng |  |
| 5 | TRIGIA | Trị giá | Number | 10 |  |  |

#### Table: CTPHIEUNHAP

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Diễn giải | Kiểu dữ liệu | Chiều dài | Miền giá trị | Ghi chú |
| 1 | MAPHIEUNHAP | Mã phiếu nhập | Char | 5 |  | Khóa ngoại đồng thời là khóa chính |
| 2 | MANVL | Mã NVL | Char | 5 |  | Khóa ngoại đồng thời là khóa chính |
| 3 | SOLUONG | Số lượng | Number | 3 | > 0 |  |
| 4 | DONGIA | Đơn giá NVL | Number | 10 | > 0 |  |
| 5 | THANHTIEN | Đơn vị tính | Number | 10 | > 0 |  |

#### Table: PHIEUXUATHANG

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Diễn giải | Kiểu dữ liệu | Chiều dài | Miền giá trị | Ghi chú |
| 1 | MAPHIEUXUAT | Mã NVL | Char | 5 |  | Khóa chính |
| 2 | NGAYLAP | Ngày lập phiếu | Date |  |  |  |
| 3 | NVLAP | Mã NV lập | Char | 5 |  | Khóa ngoại |

#### Table: CTPHIEUXUAT

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Diễn giải | Kiểu dữ liệu | Chiều dài | Miền giá trị | Ghi chú |
| 1 | MAPHIEUXUAT | Mã NVL | Char | 5 |  | Khóa ngoại đồng thời là khóa chính |
| 2 | MANVL | Tên NVL | Char | 5 |  | Khóa ngoại đồng thời là khóa chính |
| 3 | SOLUONG | Số lượng | Number | 3 | >0 |  |

#### Table: KHUYENMAI

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Thuộc tính | Diễn giải | Kiểu dữ liệu | Chiều dài | Miền giá trị | Ghi chú |
| 1 | MAKM | Mã KM | Char | 5 |  | Khóa chính |
| 2 | TENKM | Tên KM | Text | 50 |  |  |
| 3 | NGAYBD | Ngày bắt đầu | Date |  |  |  |
| 4 | NGAYKT | Ngày kết thúc | Date |  |  |  |
| 5 | LOAIKM | Loại khuyến mãi | Text | 50 |  |  |
| 6 | TRIGIAAPDUNG | Trị giá áp dụng | Number | 9 | > 0 |  |
| 7 | TRIGIAKM | Trị giá khuyến mãi | Number | 9 | > 0 |  |

## Ràng buộc toàn vẹn dữ liệu

### Trạng thái của bàn chỉ được nhận 3 giá trị (Trống, Đang phục vụ, Đang chờ)

Bối cảnh: BAN

Ɐb ϵ BAN: b.TRANGTHAI in (‘Trống’, ‘Đang chờ’, ‘Đang phục vụ’)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Thêm | Xóa | Sửa |
| BAN | + | - | +(TRANGTHAI) |



### Thành tiền của CTHD phải bằng SOLUONG\*GIABAN

Bối cảnh: CTHD

Ɐt ϵ CTHD: t.THANHTIEN = t.GIABAN\*t.SOLUONG

|  |  |  |  |
| --- | --- | --- | --- |
|  | Thêm | Xóa | Sửa |
| CTHD | + | - | +(THANHTIEN, GIABAN, SOLUONG) |

### Trị giá hóa đơn bằng tổng thành tiền của CTHD ứng với mỗi mã hóa đơn

Bối cảnh: CTHD, HOADON

Ɐt ϵ HOADON: t.TRIGIA =Σ(u ϵ CTHD Ʌ t.MAHD=u.MAHD)(THANHTIEN)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Thêm | Xóa | Sửa |
| HOADON | -(\*) | - | +(TRIGIA) |
| CTHD | + | + | +(THANHTIEN, GIABAN, SOLUONG) |

### Ngày kết thúc khuyến mãi phải lớn hơn hoặc bằng ngày bắt đầu khuyến mãi

Bối cảnh: KHUYENMAI

Ɐb ϵ KHUYENMAI: b.NGAYBD <= b.NGAYKT

|  |  |  |  |
| --- | --- | --- | --- |
|  | Thêm | Xóa | Sửa |
| KHUYENMAI | + | - | +(NGAYBD, NGAYKT) |

### Thành tiền của CTPHIEUNHAP phải bằng SOLUONG\*DONGIA

Bối cảnh: CTPHIEUNHAP

Ɐt ϵ CTPHIEUNHAP: t.THANHTIEN=t.DONGIA\*t.SOLUONG

|  |  |  |  |
| --- | --- | --- | --- |
|  | Thêm | Xóa | Sửa |
| CTPHIEUNHAP | + | - | +(THANHTIEN, DONGIA, SOLUONG) |

### Trị giá của phiếu nhập hàng phải bằng tổng THANHTIEN ứng với mỗi MAPHIEUNHAP

Bối cảnh: PHIEUNHAPHANG, CTPHIEUNHAP

Ɐt ϵ PHIEUNHAPHANG:

t.TRIGIA= Σ(u ϵ CTPHIEUNHAP Ʌ t.MAPHIEUNHAP =u.MAPHIEUNHAP)(THANHTIEN)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Thêm | Xóa | Sửa |
| PHIEUNHAPHANG | -(\*) | - | +(TRIGIA) |
| CTPHIEUNHAP | + | + | +(THANHTIEN, DONGIA, SOLUONG) |

# **Chương 3: KHÁI QUÁT MÔ HÌNH VÀ CÔNG NGHỆ SỬ DỤNG**



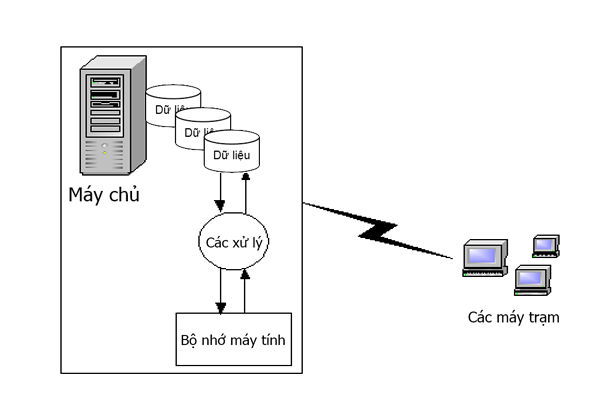
## Khái quát về hệ quản trị cơ sở dữ liệu Oracle

### Tổng quan về cơ sở dữ liệu Oracle

* Cơ sở dữ liệu Oracle (Oracle Database) là hệ thống quản lí cơ sở dữ liệu đa mô hình được sản xuất và tiếp thị bởi Tập đoàn Oracle.
* Đây là một cơ sở dữ liệu thường được sử dụng để chạy xử lý trực tuyến(OLTP), kho lưu trữ dữ liệu(DW) và khối lượng công việc cơ sở dữ liệu hỗn hợp(OLTP & DW).

### Tổng quan về kiến trúc Oracle

Oracle được kiến trúc theo mô hình 3 lớp: Lớp dữ liệu, lớp xử lý, lớp bộ nhớ.

****

Hinh, Mô hình kiến trúc 3 lớp của Oracle.

* Lớp dữ liệu (File systems): lớp dữ liệu bao gồm các tập tin dữ liệu được tổ chức lưu trữ tại các đĩa cứng của một hoặc nhiều máy chủ khác nhau(tinh phân tán của CSDL). Khi có các yêu cầu truy xuất từ phía các máy trạm, các thành phần bên trong dữ liệu đã được các xử lý bên dưới nạp đúng phần dữ liệu cần truy xuất trướ đó từ đĩa cứng vào bên trong bộ nhớ của máy chủ. Điều này giúp cho tốc độ truy xuất được hiệu quả hơn. Thông tin bên trong một CSDL Oracle sẽ có nhiều loại tập tin dữ liệu khác nhau. Tiêu biểu là một số loại tập tin sau:
* Tập tin tham số (init file): là tập tin chứa các thông tin cơ sở liên quan đến tên CSDL, vị trí các tập tin điều khiển, các tham số...
* Tập tin dữ liệu (database file): là tập tin chứa dữ liệu thật sự của CSDL
* Tập tin điều khiển (control file): là tập tin chứa các thông tin liên quan đến ngày giờ tạo CSDL, vị trí của CSDL...
* Tập tin lưu vết (redo log file): là tập tin chứa các hành động cập nhật dữ liệu (thêm, sửa, hủy) bên trong giao tác.
* Lớp xử lý bên dưới(Background Processes): là nơi đảm bảo cho mối quan hệ giữa phần CSDL vật lý và phần hiển thị trong bộ nhớ được khớp nhau.
* Lớp bộ nhớ(Memory): bao gồm nhiều thành phần khác nhau được tổ chức lưu trữ trên vùng đệm bộ nhớ của máy tính nhằm tăng tốc độ xử lý trong Oracle, khái niệm này còn được biết đến với một cái tên khác là vùng toàn cục hệ thống (System Global area).

## Khái quát về mô hình MVC

### MVC là gì?

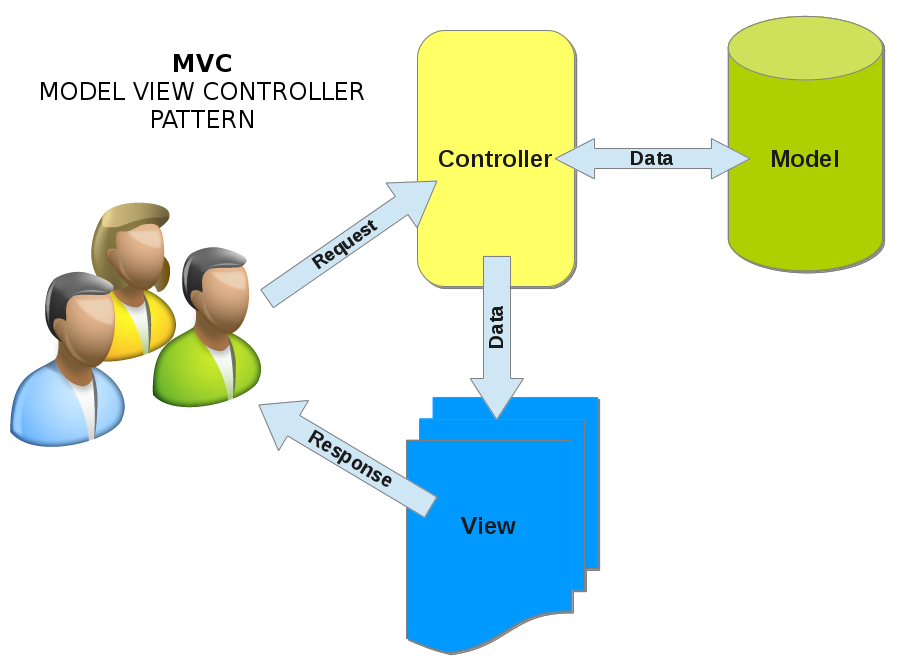
* Mô hình MVC (Model – View – Controller) là một kiến trúc phần mềm hay mô hình thiết kế được sử dụng trong kỹ thuật phần mềm. MVC là Design Pattern (mẫu thiết kế) được áp dụng rộng rãi trên hầu hết các ngôn ngữ lập trình hướng đối tượng hiện nay từ Java, C#, PHP, Visual C++… Nó giúp cho các lập trình viên tách ứng dụng của họ ra 3 thành phần khác nhau Model (Dữ liệu), View (Giao diện) và Controller (Điều khiển). Mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác.

### Các thành phần trong MVC

* **Model**: Là nơi chứa những nghiệp vụ logic, các phương thức xử lý dữ liệu, truy xuất dữ liệu từ database và gửi về **View.**
* **View**: Nơi đảm nhận việc hiển thị thông tin, tương tác với người dùng thông qua lớp này dữ liệu được truy xuất dưới database sao cho phù hợp với yêu cầu của người dùng mong muốn.
* **Controller**: Đảm bảo việc tiếp nhận yêu cầu gửi đến của người dùng thông qua lớp View đây như là một cầu nối giữa **View** và **Model**.

### Quá trình hoạt động của mô hình MVC

* **Controller – View:** Người dùng bắt đầu yêu cầu truy xuất dữ liệu thông qua tầng **View** bằng cách click vào button hoặc lựa chọn vào các Combobox,… Lớp **Controller** sẽ ghi nhận những thông tin được tầng **View** chuyển yêu cầu đến**.**
* **Controller – Model:** Sau khi tiếp nhận thông tin yêu cầu từ người dùng **Controller** thao tác xuống **Model** để lấy dữ liệu hoặc nhờ **Model** truy xuất dữ liệu khớp với yêu cầu của người dùng.
* **Model – View**: hai tầng này sẽ không được tương tác trực tiếp với nhau nhưng sẽ tương tác được thông qua lớp **Controller**, thông tin người dùng được yêu cầu tiếp nhận ở **View** sẽ nhờ **Controller** gửi tới **Model** để truy xuất dữ liệu sau đó trả về thông qua **Controller**, **Controller** chuyển đến **View,** lúc này **View** hiển thị thông tin mong muốn hay thông báo yêu cầu trả về cho người dùng.

****

Hình. Quá trình hoạt động của mô hình MVC

### Ưu và nhược điểm của mô hình MVC



#### Ưu điểm

* MVC làm cho ứng dụng trở lên trong sáng, giúp lập trình viên phân tách ứng dụng thành ba lớp rõ ràng. Điều này giúp ích rất nhiều trong việc hỗ trợ lập trình viên bảo trì hay phát triển dứng dụng thời gian lâu dài.
* MVC đang là mô hình lập trình tiên tiến bậc nhất hiện nay điều mà các framework vẫn đang nổ lực để hướng tới sự đơn giản và yếu tố lâu dài cho người sử dụng.
* Thể hiện tính chuyên nghiệp trong lập trình, phân tích thiết kế. Do được chia thành các thành phần độc lập nên giúp phát triển ứng dụng nhanh, đơn giản, dễ nâng cấp, bảo trì...

#### Nhược điểm

* Mặc dù, MVC tỏ ra lợi thế hơn nhiều so với cách lập trình thông thường. Nhưng MVC luôn phải nạp, load những thư viện đồ sộ để xử lý dữ liệu. Chính điều này làm cho mô hình trở nên chậm chạp hơn nhiều so với việc code tay thuần túy.
* MVC đòi hỏi người tiếp cận phải biết qua OOP, có kinh nghiệm tương đối cho việc thiết lập và xây dựng một ứng dụng hoàn chỉnh. Sẽ rất khó khăn nếu OOP của người sử dụng còn yếu.
* MVC tận dụng mảng là thành phần chính cho việc truy xuất dữ liệu. Nhất là với việc sử dụng active record để viết ứng dụng. Chúng luôn cần người viết phải nắm vững mô hình mảng đa chiều.
* Đối với dự án nhỏ việc áp dụng mô hình MC gây cồng kềnh, tốn thời gian trong quá trình phát triển. Tốn thời gian trung chuyển dữ liệu của các thành phần.

## Khái quát về JDBC

### JDBC là gì?

* JDBC (Java Database Connectivity) là một chuẩn API(Application Program Interface) cho phép kết nối các chương trình được viết bởi Java với nhiều hệ quản trị khác nhau ví dụ: MySQL, Oracle, MS SQL, DB2,...
* JDBC chỉ là một tập các interface, các định nghĩa, thông báo lỗi, đặc tả chứ không phải là thư viện. Với mỗi hệ quản trị cơ sở dữ liệu ta sẽ có một cài đặt JDBC riêng cho nó, ví dụ JDBC cho MySQL, JDBC cho MS SQL…

### Thành phần của JDBC

* **DriverManager**: Lớp này quản lý các Database Driver. Ánh xạ các yêu cầu kết nối từ ứng dụng Java với Data driver thích hợp bởi sử dụng giao thức kết nối phụ.
* **Driver**: Interface này xử lý các kết nối với Database Server. Hiếm khi, chúng ta tương tác trực tiếp với các đối tượng Driver này. Thay vào đó, chúng ta sử dụng các đối tượng DriverManager để quản lý các đối tượng kiểu này.
* **Connection**: Đối tượng Connection biểu diễn ngữ cảnh giao tiếp. Interface này chứa nhiều phương thức đa dạng để tạo kết nối với một Database.
* **Statement**: Chúng ta sử dụng các đối tượng được tạo từ Interface này để đệ trình các lệnh SQL tới Database. Ngoài ra, một số Interface kết thừa từ nó cung chấp nhận thêm các tham số để thực thi các thủ tục đã được lưu trữ.
* **ResultSet**: Các đối tượng này giữ dữ liệu được thu nhận từ một Database sau khi chúng ta thực thi một truy vấn SQL. Nó nóng vai trò như một Iterator để cho phép chúng ta vọc qua dữ liệu của nó.
* **SQLException**: Lớp này xử lý bất cứ lỗi nào xuất hiện trong khi làm việc với Database.

## Khái quát về JavaSwing

* Swing là một bộ công cụ tiện ích GUI cho Java. Nó là một phần của các lớp Java Foundation của Oracle - một API để cung cấp GUI cho các chương trình Java. Swing được phát triển để cung cấp một bộ các thành phần GUI tinh vi hơn bộ công cụ cửa sổ trừu tượng AWT trước đó.
* Khác nhau giữu **Swing** và **AWT:**
* Các thành phần Swing không được triển khai theo mã dành riêng cho nền tảng. Thay vào đó là viết hoàn toàn bằng Java và do đó nó độc lập với nền tảng
* Các thành phần của **Swing** là gọn nhẹ.
* Cung cấp nhiều thành phần mạnh mẽ hơn như table, list, scrollpanes, colorchooser, tabbedpane.
* **Swing** theo sau MVC để dễ dàng quản lý.

## Khái quát về Netbeans 8.2

### Giới thiệu phần mềm NetBeans.

* Đây là ứng dụng tích hợp đầy đủ tính năng để tạo ra các sản phẩm liên quan đến ứng dụng web. Nó là một công cụ hỗ trợ lập trình viết mã code miễn phí được cho là tốt nhất hiện nay. Hơn nữa, các lập trình viên phát triển Java chuyên sử dụng nó đồng thời nó có thể hỗ trợ bất kì ngôn ngữ lập trình nào. Nó được ưa chuộng bởi hỗ trợ đa ngôn ngữ khác nhau, nhiều chức năng tùy chỉnh, sửa đổi thuận tiện cho người dùng. Đây là sản phẩmcủa công tySun Microsystems đang phát triển mạnh và có gần 100 đối tác trên toàn thế giới. Ngoài ra, Netbean được viết bằng ngôn ngữ Java và có thể chạy êm trên hầu hết các hệ điều hành hiện nay như là: Windows, Linux, OS....
* NetBean IDE là môi trường phát triển rất cần thiết cho các nhà phát triển phần mềm. Bởi lẽ nó là một mã nguồn mở cung cấp các tính năng cần thiết nhất để tạo ra các ứng dụng web, thiết bị di động, desktop. Có thể khẳng định đây là phần mềm hỗ trợ soạn thảo ngôn ngữ lập trình tốt bậc nhất hiện nay. Tính năng mới cập nhật của NetBean IDE 8.2. Các chuyên viên luôn nghiên cứu không ngừng nâng cấp, cải tiến để mang tới cho người dùng những điều mới mẻ. Và đây là công cụ dành cho dân IT, nó không những hỗ trợ ECMAScript thử nghiệm 7 mà còn nâng cao HTML5/JavaScript. Đặc biệt, phần mềm này cải tiến các công cụ soạn thảo Java và Profiler cũng như nâng cao ngôn ngữ lập trình C / C ++.

### Ưu điểm phần mềm NetBeans.

* NetBeans IDE sở hữu giao diện được trực quan cho phép bạn kéo và thả các thành phần cần thiết vào một khung hình đồng thời có thể chỉnh sửa văn bản trực tiếp tại chỗ. Nó cũng là phiên bản dành cho các ngôn ngữ lập trình hiện đại như Python, Ruby, JavaScript, Groovy, C / C ++. Ngoài ra, nó tạo liên kết với các bộ máy thông tin lớn như Amazon, Facebook, Google và các trang web phổ biến hiện nay. Nhờ đó, mà các thao tác hay tác vụ được nhẹ nhàng, đơn giản hơn. Thêm vào đó, phần mềm còn cung cấp chức năng chỉnh sữa mã nguồn, hiển thị các đối tượng trong ngôn ngữ lập trình dưới dạng một pop-up. Nó cũng hỗ trợ gỡ lỗi trên máy chủ thuộc hệ thống mạng nội bộ và từ xa. NetBean IDE nay đã bổ sung các tính năng mới như tìm kiếm nhanh hay tự động biên dịch, hỗ trợ cho web framework. Từ khi ra đời cho đến bây giờ NetBean IDE đã thu hút sự quan tâm của nhiều người và là phần mềm hỗ trợ lập trình, viết code được đánh giá là tốt nhất hiện nay.
* Nếu là một lập trình viên thì trong máy tính của bạn đừng quên cài đặt nó! Vì nó rất cần thiết để phát triển các ứng dụng trên nền tảng Java đặc biệt là phù hợp trên các máy có cấu hình có RAM, CPU tương đối cao.Với lập trình PHP thì bạn hãy tải gói HTML 5 & PHP để máy tính vẫn có thể chạy êm mượt và tốc độ cao hơn, nó có gợi ý cú pháp khá tốt, và còn nhiều tính năng khá thú vị như tự động căn chỉnh mã nguồn, là môi trường phát triển không thể bỏ qua cho các lập trình viên. Để soạn thảo văn bản lập trình hiệu quả thì đây quả là sự lựa chọn thông minh, hoàn hảo.

### Các tính năng của phần mềm NetBeans.

* Dưới đây là những tính năng phổ biến mà NetBeans mang đến cho người dùng như:
* Hỗ trợ nhiều loại ngôn ngữ lập trình.
* Chỉnh sửa mã nguồn thông minh.
* Giao diện trực quan, dễ thao tác, sử dụng.
* Là một công cụ lập trình phần mềm máy tính hoặc phần mềm trên các thiết bị di động.
* Gỡ lỗi mạng nội bộ và từ xa.
* Thử nghiệm tính năng xây dựng giao diện đồ họa.
* Tính năng QuickSearch (Tìm kiếm nhanh), tự động biên dịch, hỗ trợ các Framework cho website, trình ứng dụng máy chủ GlassFish và cơ sở dữ liệu.

## Khái quát về Git

* Git là một hệ thống quản lý phiên bản phân tán được phát triển bởi Linus Torvalds vào năm 2005 ban đầu dành cho phát triển Linux. Git sử dụng mô hình phân tán, ngược lại so với SVN hoặc CSV. Mỗi nơi lưu source sẽ được gọi là Repositories, không cần lưu trữ tập trung một nơi, mà mỗi thành viên trong team sẽ có một repository ở máy của riêng mình. Có 2 dạng là Local repository và Remote repositories. Lập trình viên có thể thực hiện mọi công việc trên bản sao (local) repository trên máy tính. Sau khi hoàn thiện công việc rồi có thể đẩy lên Remote repository để chia sẻ cho người khác.
* Git có nhiều ưu điểm:
* Sắp xếp công việc tốt hơn, người dùng có để dễ dàng giải quyết từng task của mình mà không phải lo lắng cho những task liên quan.
* Linh hoạt hơn khi phải làm cùng lúc nhiều task.
* Tự tin hơn trong lúc thử nghiệm những ý tưởng mới.

## Khái quát về GitKraken

* GitKraken là giao diện hướng về phía client của Git cho các nhà phát triển phần mềm sử dụng Windows, Mac và Linux.
* GitKraken đơn giản hóa các lệnh phức tạp thành các hành động kéo và thả. Nó làm việc với các kho lưu trữ từ xa dễ dàng hơn thông qua tích hợp với GitHub, Bitbucket và GitLab. Nó cho phép bạn giải quyết xung đột hợp nhất mà không cần rời khỏi ứng dụng. Và nó hỗ trợ Gitflow, Git Hook, LFS, v.v.
* Trên hết, nó miễn phí cho mục đích phi thương mại, giáo dục và khởi nghiệp!

# **Chương 4: XÂY DỰNG VÀ QUẢN LÝ GIAO TÁC**



## Function

## Stored Procedure

## Trigger

# **Chương 5: XỬ LÝ TRUY XUẤT ĐỒNG THỜI**



## Các mức cô lập trong Oracle

## Lost Update

* **Mô tả tình huống:**
* Khi NV1 muốn sửa đổi trị giá áp dụng khuyến mãi và trị giá khuyến mãi ở phần quản lý khuyến mãi. Đồng thời ngay lúc đó, NV2 sửa đổi trị giá áp dụng khuyến mãi và trị giá khuyến mãi trên cùng một loại khuyến mãi mà NV1 thực hiện -> giá trị thay đổi của NV2 ghi đè lên giá trị thay đổi của NV1 => Lost Update.

|  |  |  |
| --- | --- | --- |
| **Session 1** | **Session 2** | **Explanation** |
| select TRIGIAAPDUNG,TRIGIAKM  from KHUYENMAI  where MAKM = 'KM001'  TRIGIAAPDUNG TRIGAIKM  ------------------------  100,000 10,000 | No action. | Session 1 truy vấn thông tin **TRIGIAAPDUNG, TRIGIAKM** ở bảng KHUYENMAI của chương trình khuyến mãi có mã là 'KM001'. |
| No action | select  TRIGIAAPDUNG, TRIGIAKM  from KHUYENMAI  where MAKM = 'KM001'  TRIGIAAPDUNG TRIGAIKM  ------------------------  100,000 10,000 | Session 2 truy vấn thông tin **TRIGIAAPDUNG, TRIGIAKM** ở bảng KHUYENMAI của chương trình khuyến mãi có mã là 'KM001'. |
| update KHUYENMAI  set TRIGIAAPDUNG = 200000,  TRIGIAKM = 20000  where MAKM = 'KM001'  --------------------------  1 row updated. | No action. | Session 1 bắt đầu Transaction 1 cập nhật giá trị **TRIGIAAPDUNG, TRIGIAKM** ở bảng KHUYENMAI của chương trình khuyến mãi có mã là 'KM001' với mức thiết lập mặc định là **READ COMMITED.**  Cập nhật giá trị thành công. |

|  |  |  |
| --- | --- | --- |
| No action. | update KHUYENMAI  set TRIGIAAPDUNG = 150000,  TRIGIAKM = 12500  where MAKM = 'KM001'  -- prompt does not return. | Session 2 cũng bắt đầu Transaction 2 cập nhật giá trị **TRIGIAAPDUNG, TRIGIAKM** ở bảng KHUYENMAI của chương trình khuyến mãi có mã là 'KM001' với mức thiết lập mặc định là **READ COMMITED.**  Nhưng dòng này đã bị khóa bởi Transaction 1. Transaction 2 sẽ thực hiện được sau khi Transaction 1 kết thúc. |
| Commit; | No action. | Transaction 1 hoàn tất công việc và kết thúc Transaction. |
| No action. | 1 row updated. | Sau khi Transaction 1 giả phóng khóa trên dòng dữ liệu **MAKM = 'KM001'** thì lúc này Transaction 2 mới thực hiện được việc cập nhật trên dòng dữ liệu đó. |
| No action. | Commit; | Transaction 2 hoàn tất công việc và kết thúc Transaction. |
| No action. | select TRIGIAAPDUNG,TRIGIAKM  from KHUYENMAI  where MAKM = 'KM001'  TRIGIAAPDUNG TRIGAIKM  ------------------------  150,000 12,500 | Session 2 truy vấn thông tin **TRIGIAAPDUNG, TRIGIAKM** ở bảng KHUYENMAI của chương trình khuyến mãi có mã là 'KM001' sau khi vừa cập nhật thành công. Kết quả khớp với dữ liệu Session 2 cập nhật. |
| select TRIGIAAPDUNG,TRIGIAKM  from KHUYENMAI  where MAKM = 'KM001'  TRIGIAAPDUNG TRIGAIKM  ------------------------  150,000 12,500 | No action. | Session 1 truy vấn thông tin **TRIGIAAPDUNG, TRIGIAKM** ở bảng KHUYENMAI của chương trình khuyến mãi có mã là 'KM001' sau khi vừa cập nhật thành công. Kết quả dữ liệu không khớp với dữ liệu Session 1 cập nhật.  => **LOST UPDATE**. |

* **Vấn đề xảy ra:**
* Giá trị do Transaction 1 cập nhật của Session 1 đã bị ghi đè bởi giá trị do Transaction 2 cập nhật của Session 2.
* **Nguyên nhân:**
* Transaction 1 và Transaction 2 đều cùng thao tác trên cùng một dòng dữ liệu, trong lúc Transaction 1 cập nhật thì không biết cũng có một Transaction khác mong muốn cập nhật dữ liệu của dòng đó. Vì vậy dẫn đến việc ghi đè dữ liệu.
* **Giải pháp:**
* Khi bắt đầu một Transaction mới mà muốn thao tác trên 1 dòng thì ta phải thiết lập lại mức cô lập cho nó. Câu lệnh thiết lập lại mức cô lập cho nó là:

'SET TRANSACTION ISOLATION LEVEL SERIALIZABLE'

* **Kết quả:**
* Nếu Transaction 1 thực hiện việc cập nhật thông tin của một chương trình khuyến mãi và Commit sau khi Transaction 2 đã bắt đầu thì Transaction 2 sẽ báo lỗi 'ORA-08177' và ROLLBACK. Sau đó Transaction 2 được thực hiện lại thì dữ liệu sẽ được cập nhật đúng với thực tế.

## Non - Repeatable Read

* **Mô tả tình huống:**
* Ban đầu quản lý xem thống kê Top 3 chi tiết mã và tên đồ uống cùng với số lượng đồ uống đã bán trong 1 năm, trước khi thống kê cho ra kết quả 1, cùng lúc đó một nhân viên bán hàng tiếp tục công việc bán hàng (có nghĩa là thêm mới một hóa đơn vào ORDERS, chi tiết hóa đơn vào CTORDERS), sau đó quản lý xuất ra file thống kê Top 3 số lượng đồ uống đã bán trong năm thì thấy kết quả 2 bị lệch so với kết quả 1. => Non-Repeatable Read.

|  |  |  |
| --- | --- | --- |
| **Session 1** | **Session 2** | **Explanation** |
| Set transaction isolation level read committed; | No action. | Session 1 thiết lập mức cô lập là **Read Commited**. |
| execute THONGKE\_TOP3(2020);  MADOUONG TENDOUONG  ------------------------  ST001 Sinh tố bơ  ST002 Sinh tố dâu  ST003 Sinh tố dưa hấu  Tong cac mon co so luong da ban nam trong Top 3 la: 3 | No action. | Session 1 truy vấn thông tin  dùng để thống kê top 3 món ưa chuộng được bán trong năm 2020 gồm có các thông tin:  mã đồ uống, tên đồ uống, tổng món top 3 ưa chuộng nhất đã bán. |
| No action. | insert into ORDERS values('SO004', TO\_DATE('15-05-2020', 'dd-MM-yyyy'), 2, 'Đã thanh toán', 24000, null, 24000);  insert into CTORDERS values('SO004', 'NN004', 12000, 2, 24000);  1 row inserted.  1 row inserted. | Session 2 bắt đầu thực hiện việc bán một loại đồ uống bằng việc thêm hóa đơn và thêm chi tiết hóa đơn vào CSDL. |
| No action. | Commit; | Transaction 2 hoàn tất công việc và kết thúc Transaction. |
| execute THONGKE\_TOP3(2020);  MADOUONG TENDOUONG  ------------------------  NN004 Number 1  ST001 Sinh tố bơ  ST002 Sinh tố dâu  ST003 Sinh tố dưa hấu  Tong cac mon co so luong da ban nam trong Top 3 la: 4 |  | Session 1 truy vấn lại lần cuối để thống kê top 3 món ưa chuộng được bán trong năm 2020 gồm có các thông tin:  mã đồ uống, tên đồ uống, tổng món top 3 ưa chuộng nhất đã bán. Nhưng dữ liệu đã bị thay đổi. => **Non-Repeatable Read** |

* **Vấn đề xảy ra:**
* Transaction 1 thực hiện truy vấn để xem thông tin top 3 loại đồ uống được khách hàng ưa chuộng của 1 năm và tổng số lượng những món đó để thống kê báo cáo. Sau đó, Transaction 2 bắt đầu thực hiện việc bán hàng của mình là thêm một hóa đơn vào bảng ORDERS và một chi tiết hóa đơn vào bảng CTORDES rồi lưu dữ liệu. Transantion 1 thực hiện thống kê lại thì thấy dữ liệu không còn giống với kết quả ban đầu.
* **Nguyên nhân:**
* Transaction 1 thiết lập mức cô lập **Read Commited** nên mỗi lần **UPDATE**, **SELECT, INSERT** trong cùng 1 thao tác dữ liệu, nó sẽ đọc lại dữ liệu từ cơ sở dữ liệu(cơ sở dữ liệu lúc này có thể bị thay đổi) mặc dù những câu lệnh này đọc trên những đơn vị dữ liệu giống nhau.
* **Giải pháp:**
* Transaction thực hiện phải thiết lập mức cô lập mới để mỗi lần **UPDATE**, **SELECT** hay **INSERT** dữ liệu không bị sai lệnh cũng như thay đổi làm cho quá trình làm việc khó khăn. Câu lệnh thiết lập lại mức cô lập cho nó là:

'SET TRANSACTION ISOLATION LEVEL SERIALIZABLE'

* **Kết quả:**
* Số lương đồ uống nằm trong Top 3 loại đồ uống được khách hàng ưa chuộng của 1 năm sẽ được giữ nguyên giống với lúc truy vấn thông tin ban đầu. Vì lúc này Transaction thưc hiện đầu tiên sẽ dùng dữ liệu trước khi Transaction sau đó thực hiện việc bán hàng(thêm hóa đơn và chi tiết hóa đơn).
* **PROCEDURE:**

create or replace PROCEDURE THONGKE\_TOP3(v\_nam NUMBER)

AS

CURSOR C

IS SELECT du.MADOUONG, du.TENDOUONG

FROM CTORDERS cto,ORDERS ord, DOUONG du

WHERE cto.MAORDER = ord.MAORDER

and du.MADOUONG = cto.MADOUONG

and EXTRACT(YEAR from ord.NGAYLAP) = v\_nam

GROUP BY du.MADOUONG, du.TENDOUONG

ORDER BY SUM(SOLUONG)DESC

FETCH FIRST 3 ROWS WITH TIES;

v\_TenDoUong DOUONG.TENDOUONG%TYPE;

v\_MaDoUong DOUONG.MADOUONG%TYPE;

v\_SumMonDaBan number(4);

BEGIN

v\_SumMonDaBan := 0;

OPEN C;

LOOP

FETCH C INTO v\_MaDoUong, v\_TenDoUong;

EXIT WHEN C%NOTFOUND;

IF C%FOUND THEN

DBMS\_OUTPUT.PUT\_LINE('Ma do uong: '||v\_MaDoUong ||' - Ten do uong: '||v\_TenDoUong);

v\_SumMonDaBan := v\_SumMonDaBan + 1;

END IF;

END LOOP;

CLOSE C;

DBMS\_OUTPUT.PUT\_LINE('Tong cac mon co so luong da ban nam trong Top 3 la: '||v\_SumMonDaBan);

END;

## Phantom Read

## DeadLock