# Mục Lục

[I. Table of Contents 2](#_Toc100671317)

[Chương I Tổng quan 3](#_Toc100671318)

[1. Database Application, ví dụ các DBA thường gặp. 3](#_Toc100671319)

[1.1 Giới thiệu 3](#_Toc100671320)

[1.2 Phân loại Database Application 3](#_Toc100671321)

[1.3 Vai trò và tầm quan trọng của Database Application 3](#_Toc100671322)

[2. Các khái niệm cơ bản 4](#_Toc100671323)

[2.1 Model 4](#_Toc100671324)

[2.2 Table 4](#_Toc100671325)

[2.3 Column 4](#_Toc100671326)

[2.4 Row 4](#_Toc100671327)

[2.5 Attribute 4](#_Toc100671328)

[2.6 Entities 4](#_Toc100671329)

[2.7 Primary Key 4](#_Toc100671330)

[2.8 Null Values 4](#_Toc100671331)

[2.9 Contraint 4](#_Toc100671332)

[2.10 Relationship types 4](#_Toc100671333)

[II. Testing Database Application 5](#_Toc100671334)

[1. Khái niệm TDA 5](#_Toc100671335)

[2. Test những gì với DB application ? 5](#_Toc100671336)

[III. Kỹ thuật kiểm chứng 5](#_Toc100671337)

[1. Transaction Flow Testing 5](#_Toc100671338)

[2. Data Flow Testing 5](#_Toc100671339)

[IV. Công cụ kiểm chứng 5](#_Toc100671340)

[1. Giới thiệu một số tool thông dụng 5](#_Toc100671341)

[V. Tài liệu tham khảo. 5](#_Toc100671342)

# Tổng quan

## Database Application, ví dụ các DBA thường gặp.

### Giới thiệu

* **Database là thuật ngữ được sử dụng nhiều trong các lĩnh vực dữ liệu, lập trình phần mềm, công nghệ thông tin, website … là thành phần vô cùng quan trọng để xây dựng và phát triển phần mềm, ứng dụng trên nền tảng mobile, PC.**
* Database (cơ sở dữ liệu) là một tập hợp những data (dữ liệu) có liên quan với nhau .  Database được duy trì dưới dạng một tập hợp các tập tin trong hệ điều hành hay được lưu trữ trong các hệ quản trị cơ sở dữ liệu.

### Phân loại Database Application

* Theo mục đích

+ Database dạng file: là dạng dữ lieejy được lưu trữ dưới dạng file. Thường được sử dụng nhất là \*.mdb, một số định dạng khác như text, ascii, \*.dbf.

+ Database quan hệ: là dạng dữ liệu thực thể, giữa các thực thể có mối liên hệ với nhau.

+ Database hướng đối tượng: là dạng dữ liệu cũng được lưu trữ dưới dạng bảng dữ liệu. Điều khác biệt là các bảng có bổ sung them các tính năng hướng đốu tượng như lưu trữ them các hành vi, nhằm thể hiện hành vi của đối tượng.

+ Database bán cấu trúc: là dạng dữ liệu được lưu dưới định dạng XML, các thông tin mô tả dữ liệu, dối tượng được trình bày trong các thẻ tag, là hướng mới trong nghiên cứu và ứng dụng về cơ sở dữ liệu.

* Theo hệ điều hành

+ Linux: MySQL, MarialDB

+ Windows: MSSQL - SQLServer

### Vai trò và tầm quan trọng của Database Application

#### Lưu trữ thông tin có hệ thống

* Cấu trúc săp xếp có hệ thống là điều làm nên sự khác biệt giữa dữ liệu hệ thống thường và cơ sở dữ liệu database.
* Dữ liệu sẽ được lưu trữ theo một cấu trúc nhất định, có tính nhất quán cao.
* Giúp người dùng thuận tiện trong việc tạo lập, lưu trữ, tìm kiếm và sử dụng dữ liệu một các chính xác, nhanh chóng.

#### Đảm bảo an toàn dữ liệu

* Đảm bảo toàn vẹn và an toàn dữ liệu là điểm quan trọng hàng đầu trong công tác lưu trữ dữ liệu.
* Tốt nhất là lưu ra file.xml hặc file.csv.

#### Đảm bảo khả năng truy xuất đồng thời của nhiều người dung trên dữ liệu.

* Nhiều người có thể sử dụng database cùng lúc mà không phải qua các khâu rườm rà phức tạp nhờ vào việc truy xuất từ các cách khác nhau.
* Thuận lợi trong việc sử dung, quản lý, truy cập dữ liệu,…

#### Linh hoạt thay đổi theo nhu caafi của người dùng.

* Có thể linh hoạt thay đổi kích cỡ và độ phức tạp của một database
* Hình thức lưu trữ database khá đa dạng như lưu dưới dạng ổ cứng, USB hay đĩa CD.

#### Công tác quản lý dễ dàng hơn.

* Database được thiết kế nhằm hỗ trợ trong việc tạo lập, cập nhật và khai thác thông tin được dễ dàng hơn.
* Dữ liệu được cập nhật thương xuyên và hoàn toàn không trùng lặp.
* Giúp tạo ra các sản phẩm chuyên nghiệp hơn, lưu trữ có hệ thống, dễ dàng trong công tác quản lý.

## Các khái niệm cơ bản

### Model

* Là một thuật ngữ dịch sang tiếng Việt gọi là Mô hình dữ liệu.
* Đề cập đến các mối quan hệ liên hợp logic và luồng dữ liệu giữa các phần tử dữ liệu khác nhau tham gia vào một môi trường thông tin chung.
* Ghi lại cách dữ liệu được lưu trữ và truy xuất.

### Table

* Là đối tượng thiết yếu trong một cơ sở dữ liệu vì chúng chứa toàn bộ thông tin hoặc dữ liệu.
* Trước khi tạo bảng cần cân nhắc yêu cầu và xác định toàn bộ các bảng có thể cần.

### Column

* Cột – là cột trong bảng thể hiện loại thông tin liên quan đến đối tượng
* Ví dụ bảng User có các cột như ID, họ tên, địa chỉ, số điện thoại,…

### Row

* Trong cơ sở dữ liệu, một hàng là một bản ghi dữ liệu trong một bảng.
* Mỗi hàng đại diện cho một hồ sơ hoàn chỉnh của dữ liệu.

### Attribute

* Thuộc tính dữ liệu – dùng để chỉ một thành phần cơ sở dữ liệu, chẳng hạn như một bảng.
* Mỗi thuộc tính sẽ thể hiện một tính chất dặc trưng riêng.

### Entities

* Thực thể là một đối tượng trong thế giới thực
* Trong lập trình, nó được xem như một lớp đối tượng.

### Primary Key

* Một khóa chính là một cột đặc biệt trong bảng cơ sở dữ liệu, được chỉ định để nhận diện tất cả hồ sơ bảng.
* Khóa chính là quan trọng đối với một cơ sở dữ liệu quan hệ hiệu quả.

### Null Values

* Là thuật ngữ được sử dụng để biểu diễn một giá trị bị khuyết thiếu.
* Giá trị NULL trong một bảng là giá trị trong một trường bị bỏ trống.
* Trường có giá trị NULL là trường không có giá trị nào.
* Trường chứa giá trị NULL khác với trường có giá trị 0 (zẻo value) và trường chứa khoảng trống (space).

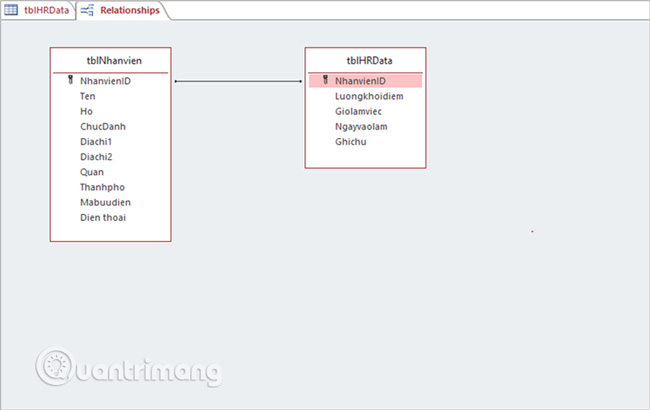
### Constraint

* Ràng buộc – là quy tắc được áp đặt cho các cột dữ liệu trên table.
* Được sử dụng để giới hạn kiểu dữ liệu nhập vào một bảng.
* Đảm bảo tính chính xác và tính đáng tin cạy cho dữ liệu trong database.

### Relationship types

* Nhằm đảm bảo CSDL không phát sinh ra những dữ liệu rác, những dữ liệu trùng lặp (tính toàn vẹn dữ liệu), thì Database thường tạo ra nhiều bảng dữ liệu - chia thông tin ra nhiều bảng - mỗi bảng này hướng quản lý một loại thông tin nào đó.
* Sau đó những bảng này kết hợp lại với nhau để có thông tin đầy đủ về đối tượng nào đó.
* Để làm điều này, bạn sẽ thiết lập những trường dữ liệu chung của các bảng từ đó hình thành mối liên hệ (relationship) giữa các bảng.

#### + One – One (1-1)



Mối quan hệ một - một (1-1)

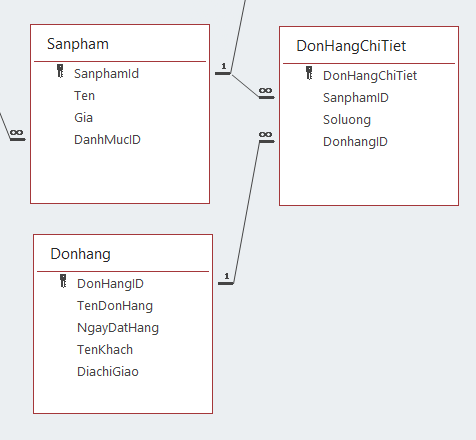
* Được sử dụng để liên kết một bản ghi từ một bảng với một và chỉ một bản ghi trong một bảng khác.
* Mối quan hệ 1-1 được tạo ra khi cả hai trường kết nối là khóa chính hoặc có các rang buộc duy nhất.

#### + One -Many (1-n)

Mối quan hệ một - nhiều (1-n)

* Một dòng của bảng một liên kết với nhiều dòng của bảng nhiều.
* Mối quan hệ 1-n được hình thành bằng cách tạo liên hệ giữa khóa chính (Primary key) và khóa ngoại (Foreign key).

#### + Many - Many



Mối quan hệ nhiều - nhiều (n-n)

* Một sản phẩm có thể nằm trong nhiều đơn hàng, ngược lại một đơn hàng có thể có nhiều sản phẩm. Từ đó hai bảng này hình thành mối quan hệ nhiều nhiều. Mối liên hệ này được hình thành qua hai khóa ngoại của*DonHangChiTiet* là có *DonHangID* và *SanphamID.*
* Mối quan hệ n-n được hình thành do mối quan hệ 2 khóa ngoại (Foreign key).

# Testing Database Application

## Khái niệm.

* Là quá trình kiểm tra độ chính xác và tính toàn vẹn của cơ sở dữ liệu.
* Đảm bảo rằng các dữ liệu là chính xác và duy nhất
* Có 2 loại Database Testing:

+ Kiểm thử dữ liệu thực (Actual Data Testing): là kiểm tra nội dung dữ liệu có chính xác hay không.

+ Kiểm thử rang buộc dữ liệu (Data Integrity Testing): là các hoạt động kiểm thử nhằm đảm bảo dữ liệu không bị hỏng, các Schema dữ liệu là đúng đắn và các chức năng của các ứng dụng CSDL hoạt động đúng.

## Test những gì với DB application ?

### Integrity Testing (Kiểm thử ràng buộc toàn vẹn trên CSDL)

### Control Testing

#### Security Testing (kiểm chứng khả năng bảo vệ từ các truy cập bất hợp lệ)

#### Backup Tesing (kiểm tra khả năng sao lưu dữ liệu)

#### Recovery Testing (Kiểm tra khả năng phục hồi dữ liệu)

### Transaction Concurency Testing

#### Kiểm chứng các thao tác trên dữ liệu vào cùng một thời điểm.

#### Khả năng xuất hiện nhiều lỗi nghiêm trọng.

# Kỹ thuật kiểm chứng

## Transaction Flow Testing

a. Giới thiệu về Transaction:

- Transaction là một đơn vị công việc được nhìn nhận từ quan điểm của người dùng hệ thống.

- Một Transaction bao gồm một chuỗi các hoạt động, một số trong số đó được thực hiện bởi một hệ thống, con người hoặc thiết bị bên ngoài hệ thống.

\*Ví dụ: Bạn có 2 tài khoản ngân hàng A và B, bạn chuyển 1.000.000đ từ tài khoản A vào tài khoản B.

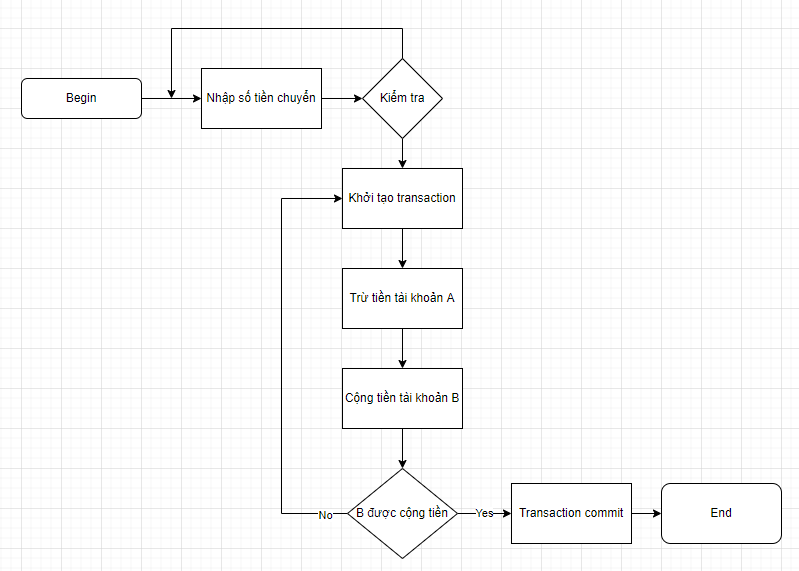
Transaction ở đây sẽ gồm 2 hành động:

Tài khoản A bị trừ 1.000.000đ

Tài khoản B cộng thêm 1.000.000đ

Bây giờ A bị trừ 1.000.000đ nhưng vì 1 lý do nào đấy B bị lỗi không nhận được 1.000.000đ thì transaction sẽ bị hủy, số tiền của 2 tài khoản A và B sẽ không bị thay đổi.

b. Transaction flow Graphs (Đồ thị luồng Transaction):



c. Các bước thực hiện Transaction Flow Testing:

- Bước 1: Xác định Transaction Flow và vẽ Transaction Flow Graph (Lưu ý: dựa trên yêu cầu người dung, không dựa trên thiết kế).

- Bước 2: Xác định định tất cả các đường đi trong Graph (tương tự như phủ nhánh và phủ điều kiện).

- Bước 3: Thiết kê Tese Case.

d. Ứng dụng:

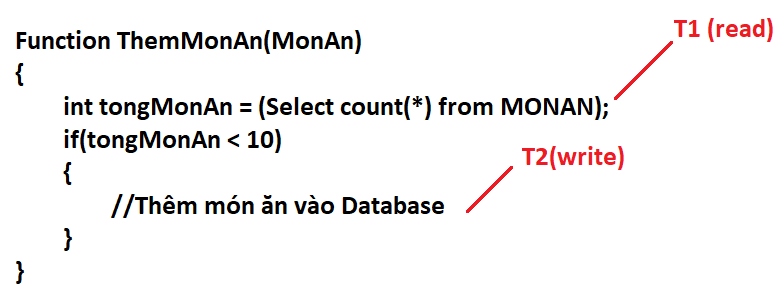
- Dùng trong Black Box Testing.

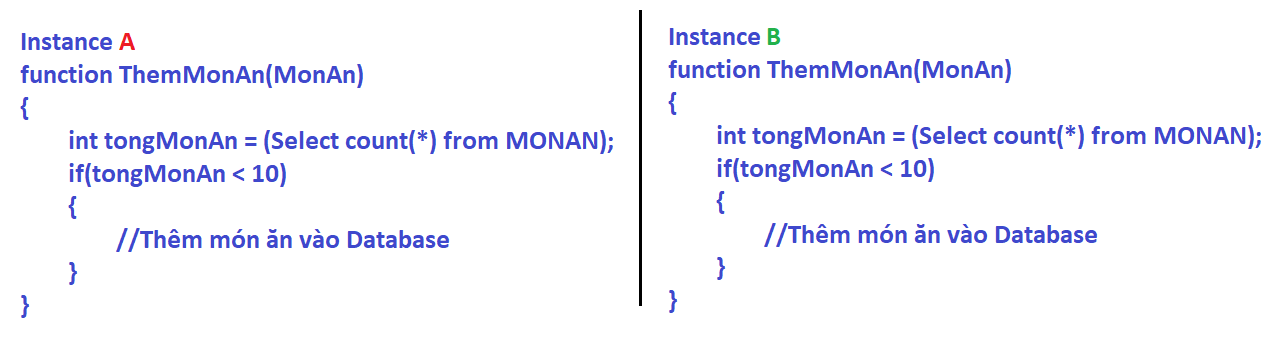
- Giúp tìm được những lỗi nghiêm trọng trong hệ thống.

- Là nền tảng dung để test các chức năng của hệ thống.

## Data Flow Testing

a. Ví dụ:





\*Thực thi: Trên một thể hiện (instance) của ứng dụng không có vấn đề. Nhưng hai thể hiện của ứng dụng cùng chạy đồng thời theo Schedule sau thì sẽ gây ra lỗi:

\*Schedule: T1(A) -> T1(B) -> T1(A) -> T2(B)

+ Instance A : read tongMonAn = 9

+ Instance B : read tongMonAn = 9

+ Instance A : write them món ăn vào Database (lúc này tongMonAn = 10)

+ Instance B : write them món ăn vào Database (lúc này tongMoAn = 11)

-> Kết luận: tongMonAn sau khi thực hiện hai thể hiện đồng thời sẽ có tongMonAn = 11 vượt quá mức kiểm tra quy định trước đó (tongMonAn < 10) gây ra lỗi.

b. Bảng phân tích kiểm chứng Data Flow

- T(i), T(j): là các thao tác đọc ghi trên dữ liệu của một thể hiện A.

- T(k) là một thao tác nằm giữa T(i) và T(j) trên cũng một dữ liệu của thể hiện B.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T(i)** | Read | Read | Read | Read | Write | Write | Write | Write |
| **T(k)** | Read | Read | Write | Write | Read | Read | Write | Write |
| **T(j)** | Read | Write | Read | Write | Read | Write | Read | Write |
| **Error** | No | No | Yes | Yes | No | Yes | Yes | Yes |

- Xác định dãy các thao tác đọc ghi dữ liệu.

- Xác định các Schedule dạng <TA(i), TB(k), TA (j)>.

- Với mỗi Schedule : xác định trạng thái dữ liệu qua các bước thực hiện Schedule.

b. Ứng dụng:

- White Box Testing.

- Phân tích lỗi dựa trên trạng thái của dữ liệu thông qua các hoạt động Read, Write.

- Test transaction Concurency (kiểm tra Transation đồng thời).

# Công cụ kiểm chứng

## Giới thiệu một số tool thông dụng

# Tài liệu tham khảo.

[1]. \_TKhao\_Testing Applications on the Web.pdf

[2]. Refs.DBApp.TestingDB.pdf

[3]. <https://viblo.asia/newest>

[4]. <https://wiki.tino.org/>

[5]. <https://viblo.asia/p/tim-hieu-ve-database-testingphan-1-63vKj2jVK2R>

[6]. <https://www.tutorialspoint.com/database_testing/database_testing_overview.htm>

[7]. <https://www.youtube.com/channel/UC46vj6mN-6kZm5RYWWqebsg>