

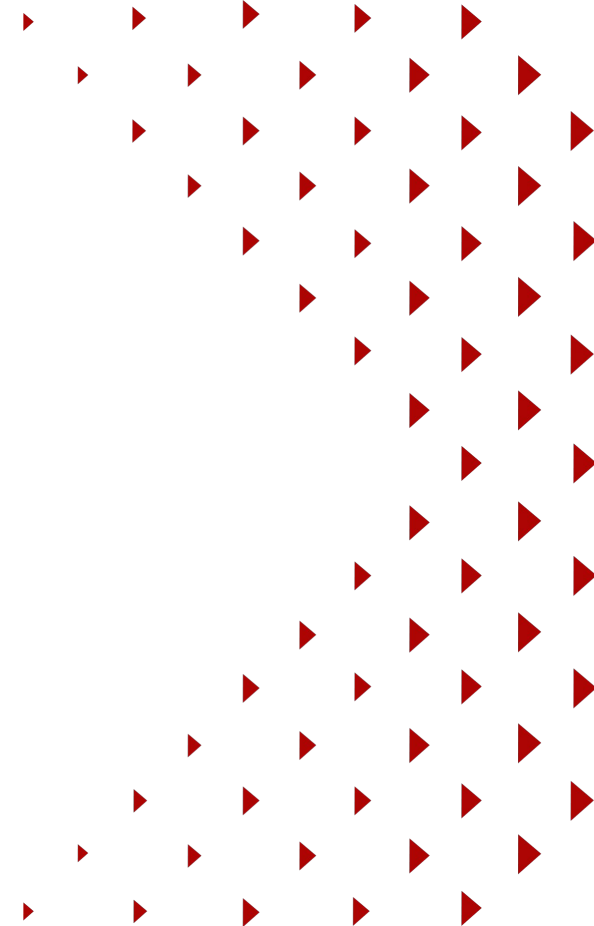


BÀI 5:

Index – View

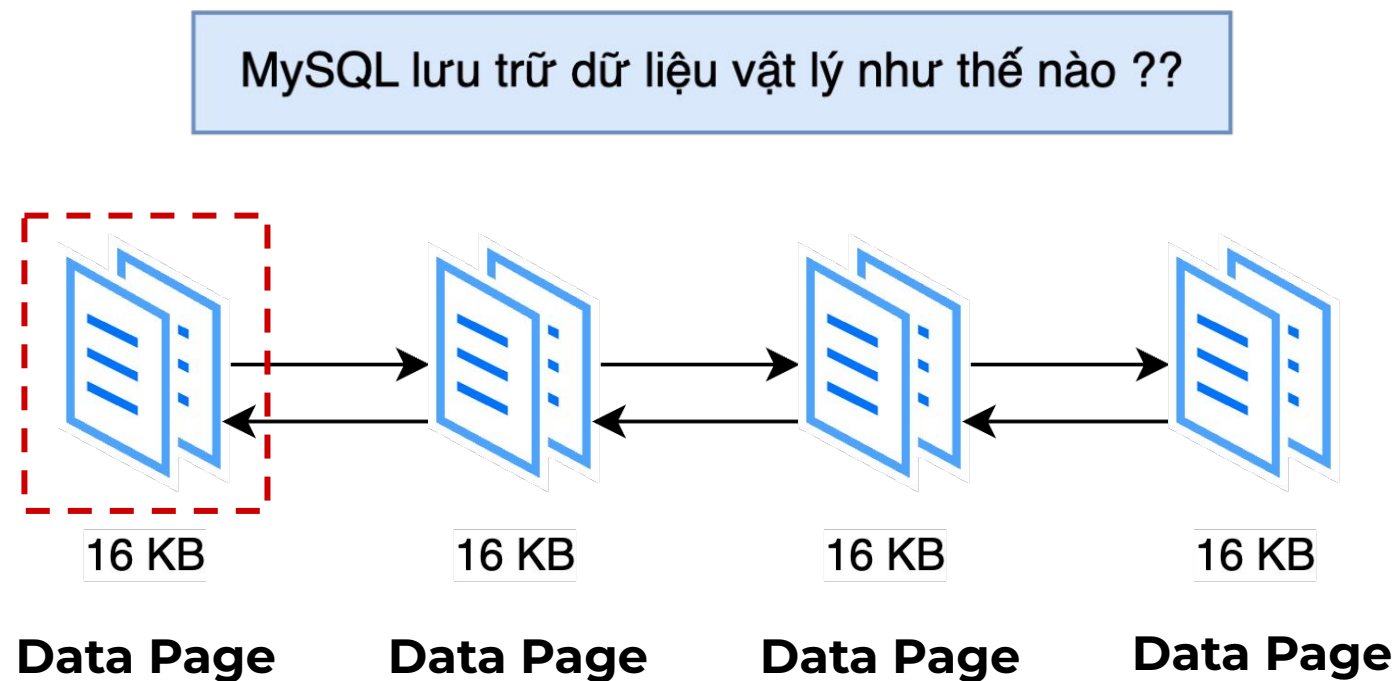
Module: Fundamental Database

Phiên bản: 1.0

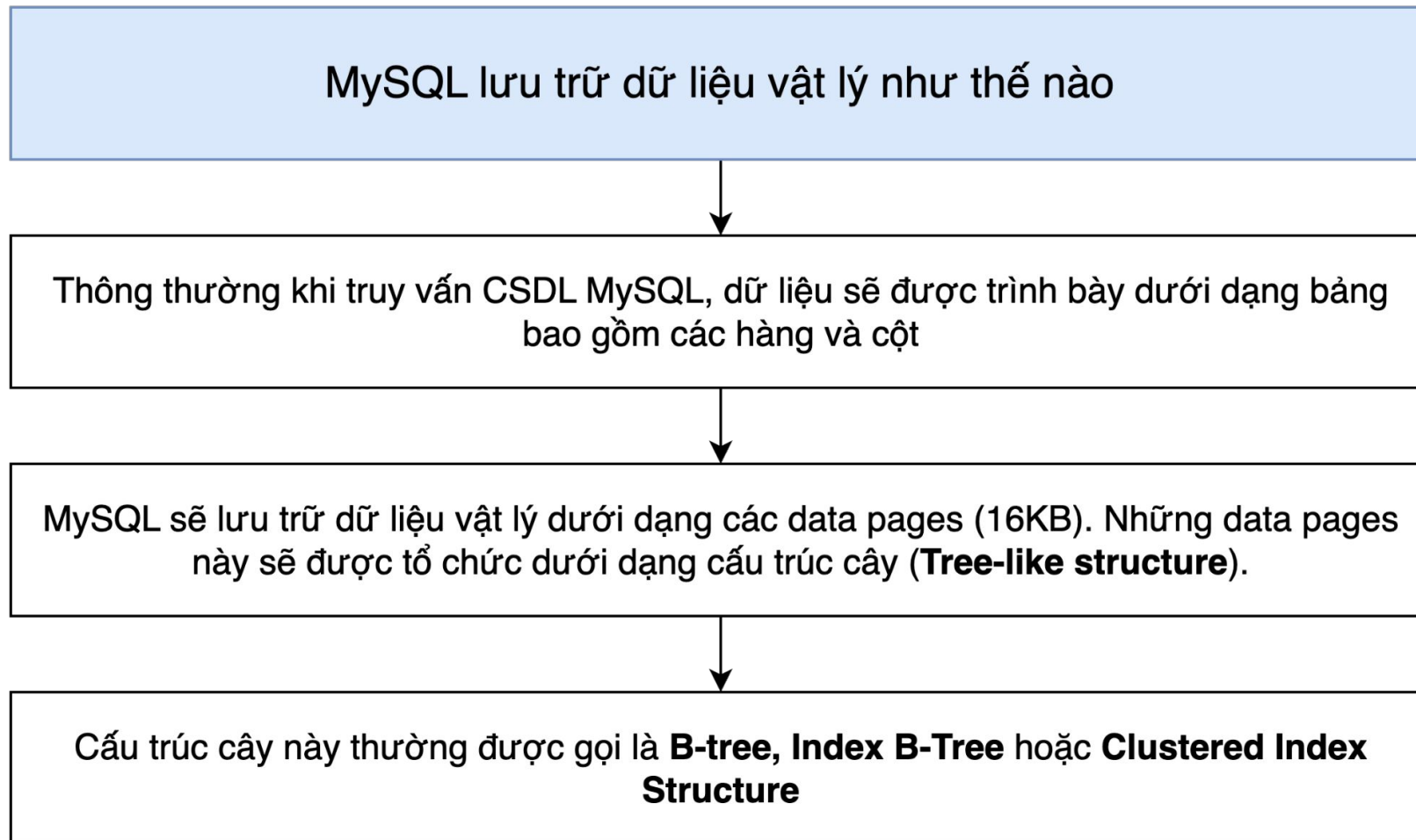


1. **MySQL lưu trữ dữ liệu vật lý như thế nào ?**
2. **Tổng quan về index chỉ mục trong MySQL**
3. **Thao tác với index (chỉ mục)**
4. **View trong MySQL**

1. MySQL lưu trữ dữ liệu vật lý như thế nào?



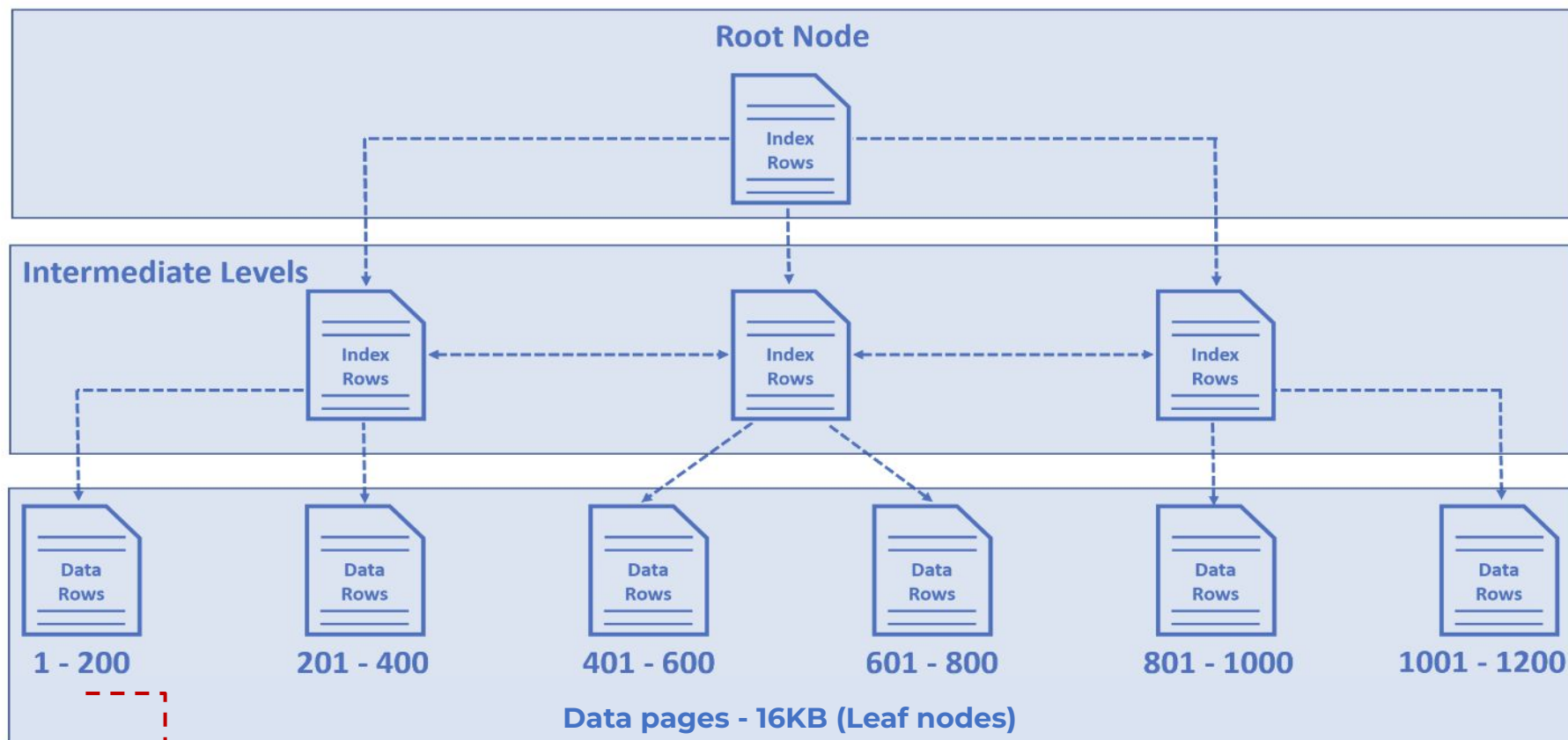
1. MySQL lưu trữ dữ liệu vật lý như thế nào?



1. MySQL lưu trữ dữ liệu vật lý như thế nào?

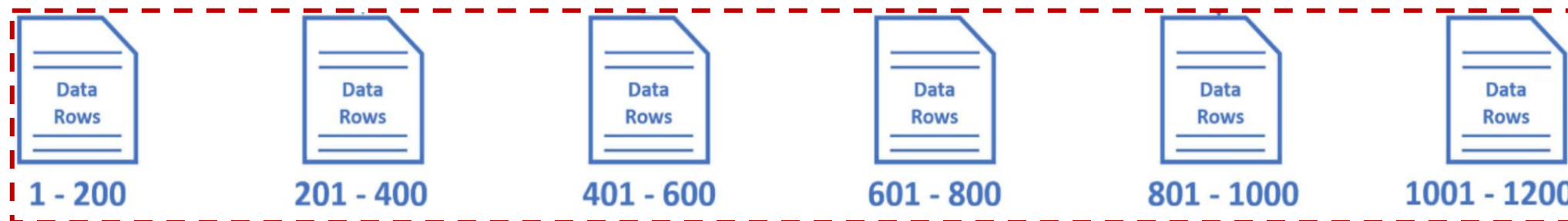
Employees Table		
Employeeid (Primary Key)	Name	Email
1	Peter Parker	peter@avengers.com
2	Tony Stark	tony@avengers.com
3	Steve Rogers	steve@avengers.com
...
1200	Natasha Romanoff	natasha@avengers.com

1. MySQL lưu trữ dữ liệu vật lý như thế nào?



Số lượng dòng (data) được lưu trữ trong từng data page sẽ tùy thuộc vào kích cỡ (Size) của mỗi dòng

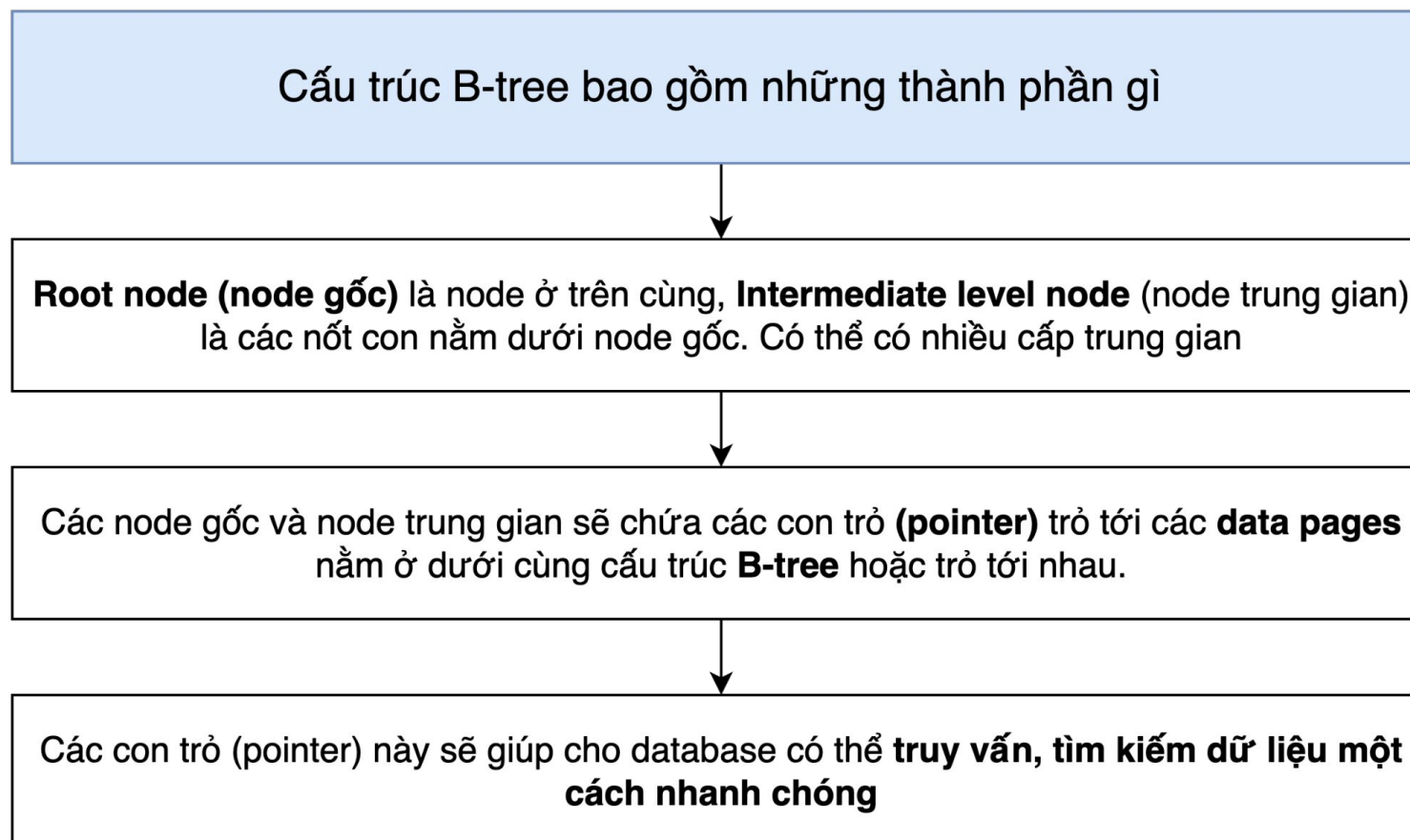
1. MySQL lưu trữ dữ liệu vật lý như thế nào?



Mặc định, khi một bảng được tạo ra có một cột là khoá chính (Primary key) thì khoá chính sẽ mặc định được coi là **clustered index**

Điều này đơn giản là các hàng trong bảng sẽ được sắp xếp theo thứ tự của khoá chính hay clustered index đó

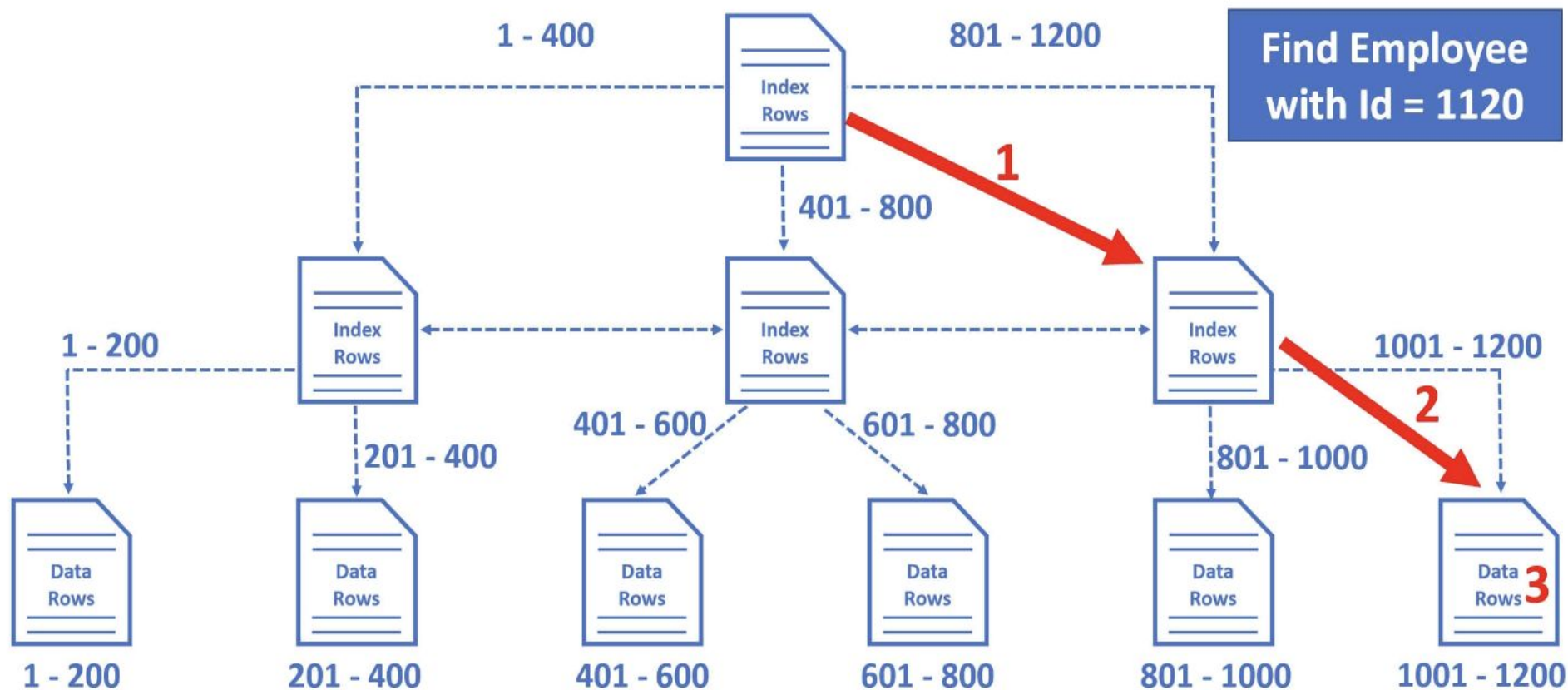
1. MySQL lưu trữ dữ liệu vật lý như thế nào?



1. MySQL lưu trữ dữ liệu vật lý như thế nào?

```
Select * from Employees where EmployeeId = 1120
```

1. MySQL lưu trữ dữ liệu vật lý như thế nào?



1. MySQL lưu trữ dữ liệu vật lý như thế nào?

Chỉ với 3 bước, đã có thể tìm ra bản ghi với id = 1120

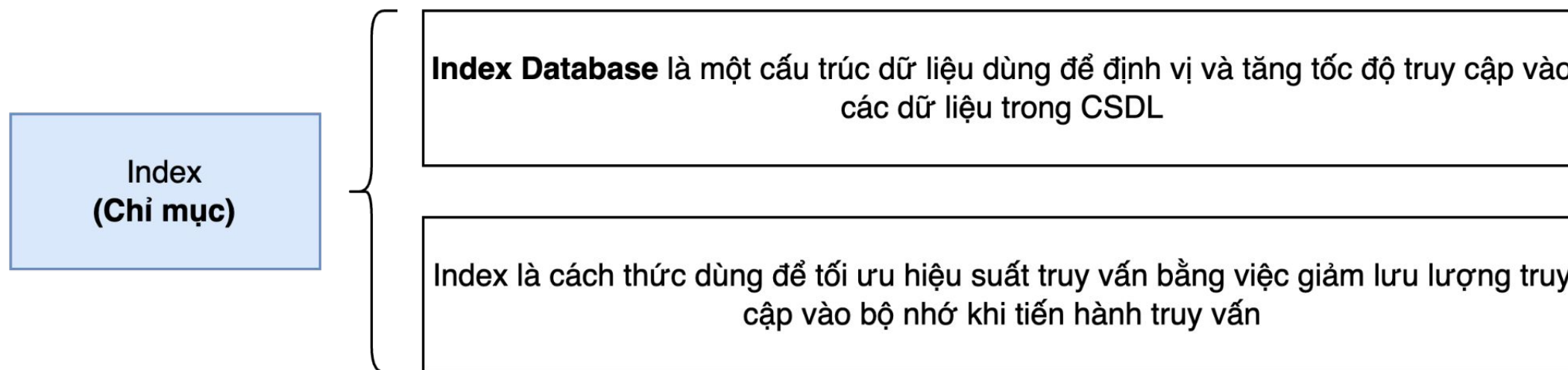
→ Dù có hàng triệu bản ghi, database engine vẫn có thể dễ dàng tìm ra bản ghi với **clustered index id**

→ Nếu tiến hành tìm kiếm với các trường khác PK ???

2. Tổng quan index (Chỉ mục) trong MySQL

CONTENTS	
Foreword	xiv
Preface	xvii
Acknowledgments	xxii
Introduction: A Dream Within Reach	xxvi
PART I • BEYOND OBSERVATION	1
Chapter One Space Becomes an Industry	2
Chapter Two Post-Apollo	6
PART II • TADPOLES, UNICORNS, AND ZEBRAS	11
Chapter Three New Energy, New Space	12
Chapter Four From Evolution to Revolution	14
Chapter Five Successful Endeavors	18
Chapter Six New Paradigms	29
PART III • THE CASE FOR NEWSPACE	37
Chapter Seven Space Is Open for Business	38
Chapter Eight Lower Costs, Lower Barriers	43
Chapter Nine A Data-Driven World	61
Chapter Ten Off-World Opportunities	80
Chapter Eleven The International Space Station	109
Chapter Twelve Space Can Do Quadrillions	119
PART IV • INVESTING IN THE COSMOS	125
Chapter Thirteen An Industry in Its Own League	126
Chapter Fourteen Increased Investments	131
Chapter Fifteen Signals for Success	141
PART V • JOINING THE MOVEMENT	145
Chapter Sixteen Moonshots	146
Chapter Seventeen Failures and the Future	154
Chapter Eighteen Better Business Models	166
Chapter Nineteen Space Is Not a Monolith	178
PART VI • POLICY AND POLITICS	181
Chapter Twenty Global Developments	182
Chapter Twenty-One The Regulatory Landscape	190
Chapter Twenty-Two International Issues	205
Chapter Twenty-Three The Space Rush	213
PART VII • THE BLUEPRINT OF EVOLUTION	217
Chapter Twenty-Four Space Is Everywhere	218
Chapter Twenty-Five Inspiring the Future	222
Chapter Twenty-Six Sci-Fi and Society	240
Chapter Twenty-Seven Imagining New Economies	243
Chapter Twenty-Eight Aspiring to Evolve	246
EX ASTRIS, AD ASTRA	249
Afterword	250
Insights From Our Sponsors	252
Case Study Planet	267
Enabling a Trillion-Dollar Space Industry	270
Reader Resources: Breaking Into NewSpace	289
Glossary	298
References	328
Index	374
About the Author	389

2. Tổng quan index (Chỉ mục) trong MySQL



2. Tổng quan index (Chỉ mục) trong MySQL

Clustered Index

Non-clustered Index

2. Tổng quan index (Chỉ mục) trong MySQL

Clustered Index

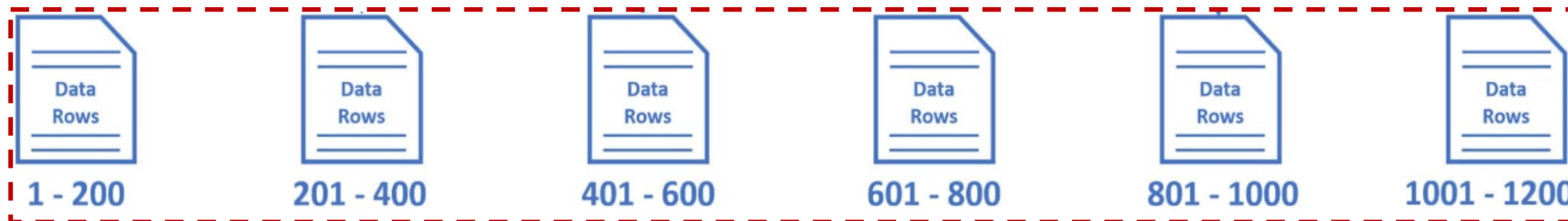
Trong MySQL, **clustered index** là một dạng chỉ mục đặc biệt trong đó dữ liệu của bảng được lưu trữ theo thứ tự sắp xếp của chỉ mục này

Clustered index giúp tối ưu hóa hiệu suất khi truy vấn và tìm kiếm dữ liệu trong bảng, vì dữ liệu thực tế được sắp xếp vật lý theo thứ tự của chỉ mục.

Trong MySQL (với InnoDB), chỉ mục được tạo mặc định trên khóa chính (PRIMARY KEY) sẽ là clustered index.

Nếu bảng không có PRIMARY KEY, InnoDB sẽ tự động chọn một chỉ mục duy nhất khác hoặc tạo một clustered index nội bộ nếu không có chỉ mục nào phù hợp.

2. Tổng quan index (Chỉ mục) trong MySQL



Sắp xếp theo thứ tự tăng dần

2. Tổng quan index (Chỉ mục) trong MySQL



Truy vấn toàn bộ customers có City là London

```
SELECT * FROM Customers  
WHERE City LIKE 'London';
```

Khi tiến hành truy vấn toàn bộ customers có city là London

→ Truy vấn phải quét qua toàn bộ row trong bảng Customers

→ Nếu có hàng trăm nghìn bản ghi → Thời gian truy vấn sẽ rất lâu

2. Tổng quan index (Chỉ mục) trong MySQL

Non-clustered Index

Non-clustered index trong MySQL là một loại chỉ mục lưu trữ các giá trị của cột chỉ mục cùng với các con trỏ (hoặc tham chiếu) đến các vị trí dữ liệu trong bảng thay vì sắp xếp dữ liệu theo thứ tự của chỉ mục như clustered index

Non-clustered index rất hữu ích khi bạn cần tăng tốc độ truy vấn trên các cột không phải là khóa chính.

Non-clustered index chứa các giá trị của cột chỉ mục cùng với các tham chiếu đến các bản ghi dữ liệu.

Có thể tạo nhiều non-clustered index trên các cột khác nhau trong một bảng, giúp tối ưu hóa cho nhiều loại truy vấn khác nhau.

2. Tổng quan index (Chỉ mục) trong MySQL

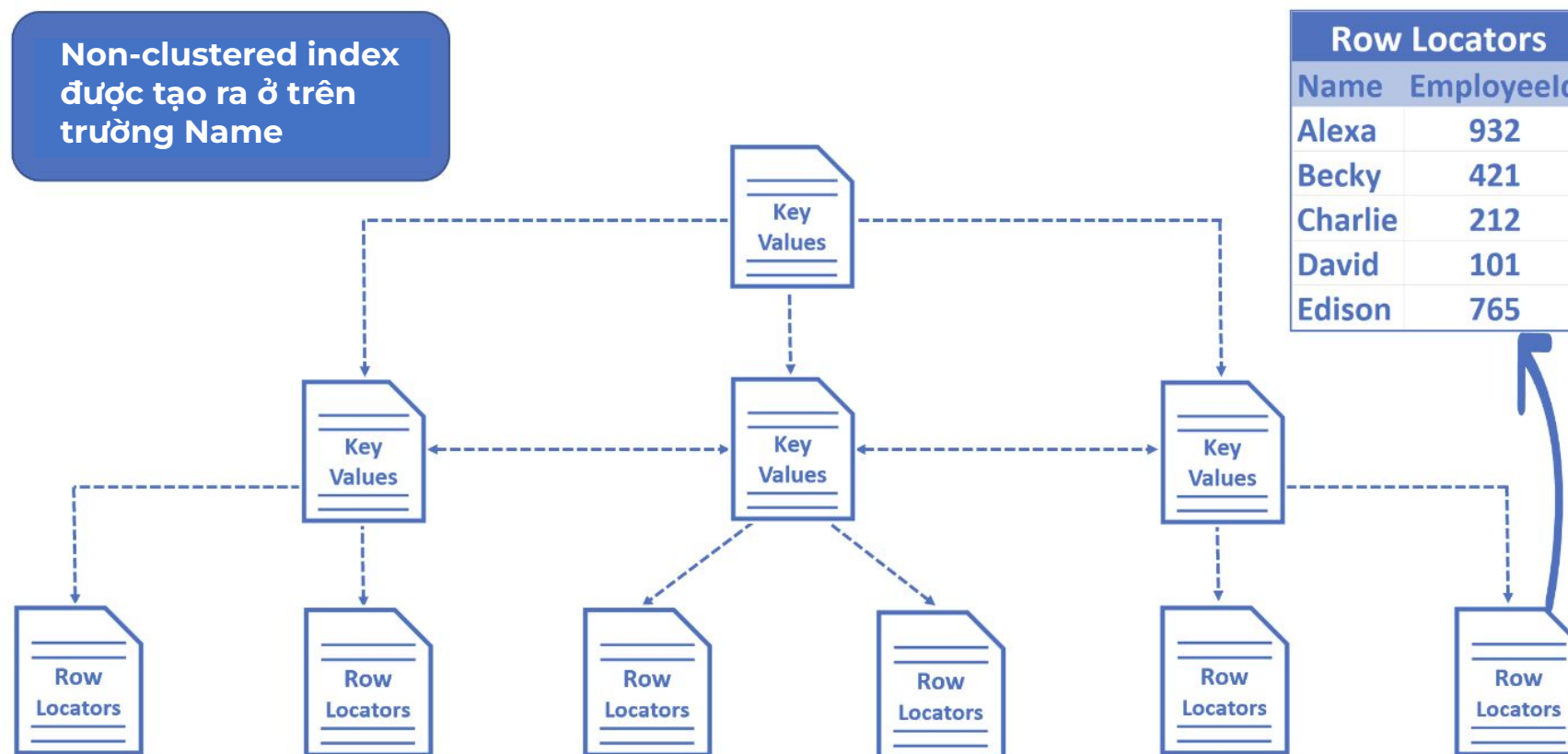
```
Select * from Employees where Name = 'Peter Parker'
```

2. Tổng quan index (Chỉ mục) trong MySQL

Employees Table		
EmployeeId (Primary Key)	Name	Email
1	Peter Parker	peter@avengers.com
2	Tony Stark	tony@avengers.com
3	Steve Rogers	steve@avengers.com
...
1200	Natasha Romanoff	natasha@avengers.com

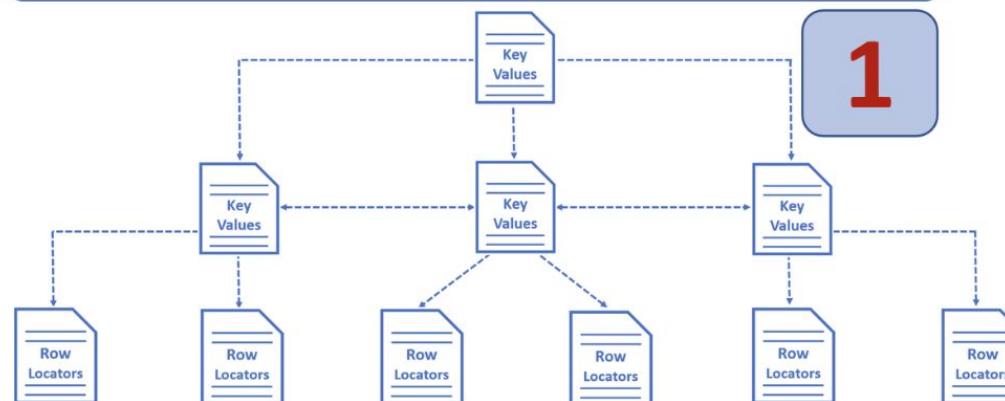
Tạo một chỉ mục Non-clustered Index cho trường name

2. Tổng quan index (Chỉ mục) trong MySQL



2. Tổng quan index (Chỉ mục) trong MySQL

Chỉ mục non-clustered index được tạo ra ở trường name



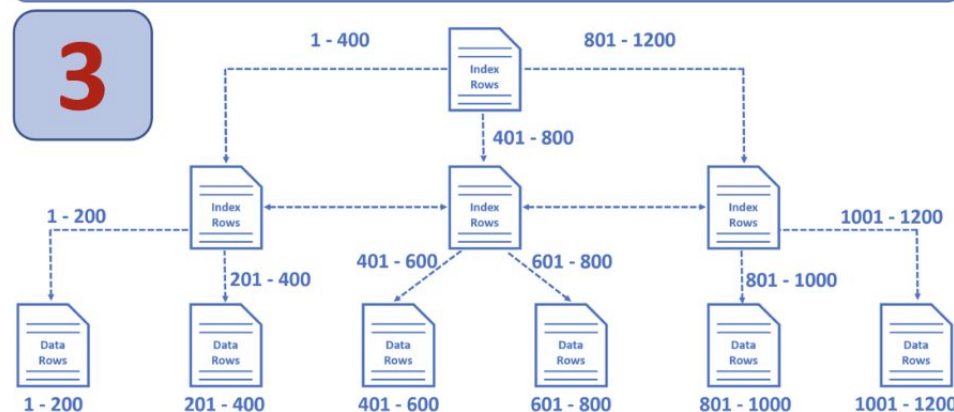
Row Locators bao gồm name đã được sắp xếp (a-z) và id

Row Locators	
Name	EmployeeId
Alexa	932
Becky	421
Charlie	212
David	101
Edison	765

2

2. Tổng quan index (Chỉ mục) trong MySQL

Chỉ mục clustered index được tạo ra ở trường id



Dữ liệu trong data rows được sắp xếp bởi trường id

4

Employees Table		
Employeeid (Primary Key)	Name	Email
1	Peter Parker	peter@avengers.com
2	Tony Stark	tony@avengers.com
3	Steve Rogers	steve@avengers.com
...
1200	Natasha Romanoff	natasha@avengers.com

2. Tổng quan index (Chỉ mục) trong MySQL

Phân tích các câu lệnh truy vấn

```
EXPLAIN SELECT * FROM worker;
```

```
EXPLAIN ANALYZE SELECT * FROM worker;
```


3. Thao tác với index (chỉ mục) - Tạo mới

```
CREATE INDEX index_name  
ON table_name (column_list)
```

3. Thao tác với index (chỉ mục) - Liệt kê

```
SHOW INDEXES FROM table_name;
```

3. Thao tác với index (chỉ mục) - Xóa

```
DROP INDEX index_name  
ON table_name;
```

4. View trong MySQL

View

View (Bảng ảo) trong MySQL là một đối tượng trong cơ sở dữ liệu đại diện cho một truy vấn SQL đã được lưu trữ. View không chứa dữ liệu thực tế mà chỉ lưu trữ câu lệnh truy vấn để lấy dữ liệu từ các bảng khác

Khi truy vấn dữ liệu từ một view, MySQL sẽ thực thi câu lệnh truy vấn của view và trả về kết quả từ các bảng liên quan.

View giúp đơn giản hóa các truy vấn phức tạp và hạn chế quyền truy cập trực tiếp vào dữ liệu nhạy cảm bằng cách chỉ hiển thị các cột hoặc hàng cần thiết.

View có thể được sử dụng để giữ cho các truy vấn phức tạp nhất quán, đặc biệt khi các bảng cơ sở được thay đổi.

4. View trong MySQL - Tạo View

```
CREATE VIEW view_name AS  
SELECT columns  
FROM tables  
WHERE conditions;
```

4. View trong MySQL - Sử dụng View

Có thể truy vấn view giống như truy vấn bảng thông thường

```
SELECT * FROM view_employee_names;
```

4. View trong MySQL - Cập nhật View

Trong một số trường hợp, bạn có thể sử dụng view để cập nhật dữ liệu trong bảng gốc, nhưng view phải thỏa mãn một số điều kiện nhất định

- View phải dựa trên một bảng duy nhất (Không có JOIN)
- Không sử dụng các từ khóa GROUP BY, DISTINCT, hoặc HAVING
- Không sử dụng các Aggregate function
- Không sử dụng UNION
- Không sử dụng Sub query trong câu truy vấn
- Không sử dụng hàm hoặc các biểu thức phức tạp trong cột
- Không sử dụng LIMIT

4. View trong MySQL - Cập nhật View

```
UPDATE view_managers  
SET salary = salary + 1000  
WHERE emp_id = 101;
```


4. View trong MySQL - Xoá View

```
DROP VIEW view_employee_names;
```

- ❑ **Nắm được cách MySQL lưu trữ dữ liệu vật lý như thế nào?**
- ❑ **Nắm được cách hoạt động của index trong MySQL**
- ❑ **Nắm được cách sử dụng View trong MySQL**



KẾT THÚC

HỌC VIỆN ĐÀO TẠO LẬP TRÌNH CHẤT LƯỢNG NHẬT BẢN