

## Bài 7: Ngôn ngữ truy vấn đa chiều (MDX - Multi-Dimensional eXpressions)

---

---

---

---

---

---

---

### Nội dung

- Hiểu khái niệm của MDX
- Cấu trúc cơ bản của câu truy vấn MDX
- Sử dụng câu truy vấn dữ liệu trên khối dữ liệu
- Một số hàm trong MDX

---

---

---

---

---

---

---

### Giới thiệu

- SQL (Structure Query Language) là ngôn ngữ sử dụng để truy xuất dữ liệu từ CSDL quan hệ.
- Ngôn ngữ MDX (Multi Dimension eExpress) là dạng mở rộng của ngôn ngữ truy vấn SQL sử dụng truy vấn dữ liệu đa chiều

---

---

---

---

---

---

---

## Cấu trúc cơ bản của ngôn ngữ truy vấn MDX

- Áp dụng MDX để khai thác kho dữ liệu OLAP với khối đa chiều nhằm hỗ trợ cho việc tổng hợp thông tin và ra quyết định.
- Về cấu trúc: giống cấu trúc của một câu truy vấn SQL trên CSDL thông thường, nhưng mở rộng hơn để truy vấn trên khối dữ liệu đa chiều.

---

---

---

---

---

---

---

## SQL vs. MDX

- |                          |                       |
|--------------------------|-----------------------|
| • Select, From, Where    | • Select, From, Where |
| • Kết quả hai chiều      | • Kết quả đa chiều    |
| • Chọn các cột           | • Chọn các tập ô      |
| • Kết quả là một quan hệ | • Kết quả là một khối |
| • Chọn từ các quan hệ    | • Chọn từ một khối    |

---

---

---

---

---

---

---

## Câu truy vấn đa chiều (MDX)

```
[WITH  
  MEMBER ...  
  SET ...]  
SELECT  [ <mô tả về chiều thứ 1> ]  
        [ , <mô tả về chiều thứ 2>, ... ]  
FROM    [ <khối dữ liệu> ]  
[WHERE [ <mô tả về các điều kiện cắt lớp> ]]
```

---

---

---

---

---

---

---

## Câu truy vấn đa chiều (MDX) (2)

- Trong MDX:
  - With: dùng để định nghĩa thêm chiều mới từ các chiều đã tồn tại
  - Phát biểu SELECT được dùng để mô tả một tập dữ liệu là một tập con của dữ liệu đa chiều.
  - Một câu truy vấn MDX có chứa một mệnh đề SELECT, một mệnh đề FROM và một mệnh đề tùy chọn WHERE.

---

---

---

---

---

---

---

## Quy ước về cú pháp MDX

- Hàm
  - Một số sử dụng ký hiệu hàm theo phong cách VB
    - Ví dụ: hàm Order()
  - Một số sử dụng định dạng đối tượng
    - Ví dụ: BoMon.Members – Members là hàm
- Thứ tự trục
  - COLUMNS, ROWS, PAGES, CHAPTERS, SECTIONS
  - Mỗi trục được đánh số thứ tự bắt đầu từ 0
    - Ví dụ: { [Dim\_ThoiGian].Members } on Axis(0)

---

---

---

---

---

---

---

## Quy ước về cú pháp MDX (2)

- Dấu phẩy – xây dựng một tập hợp bằng cách liệt kê các các bộ
  - Ví dụ: {[Nam].[2001], [Nam].[2002], ...}
- Dấu hai chấm – xây dựng một tập hợp bằng cách chỉ định một phạm vi
  - Ví dụ: {[Nam].[2001] : [Nam].[2005] }
- .Members – trả về tập tất cả các thành viên
  - Ví dụ: [BoMon].Members

---

---

---

---

---

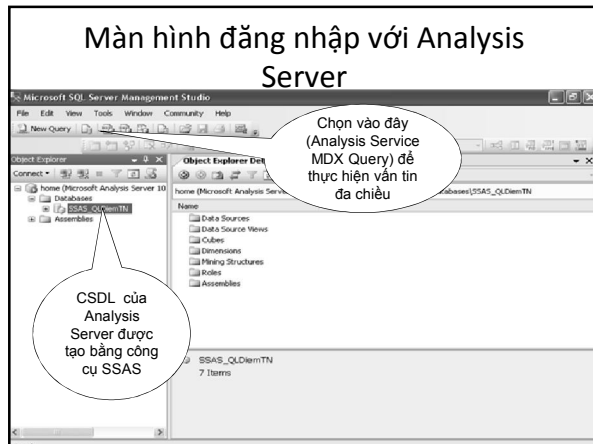
---

---

## Demo – Viết câu lệnh MDX Query

## Khởi dữ liệu điểm tốt nghiệp

## Chọn kết nối đến Analysis Server




---

---

---

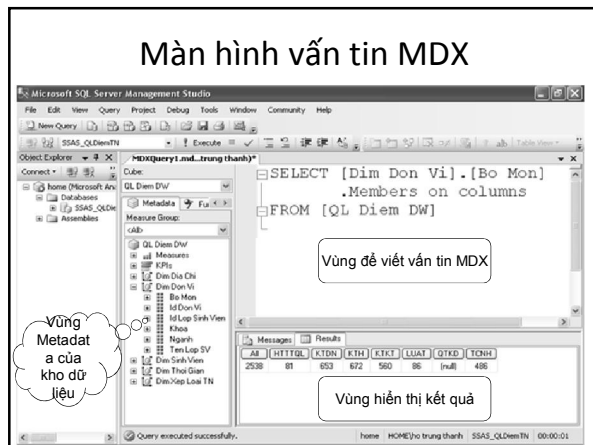
---

---

---

---

---




---

---

---

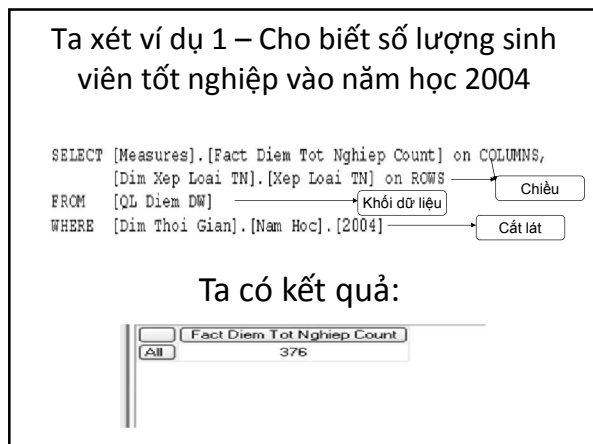
---

---

---

---

---




---

---

---

---

---

---

---

---

### Giải thích ví dụ 1

- Mệnh đề SELECT xác định các chiều của tập kết quả của truy vấn MDX, trong ví dụ là 1 chiều COLUMNS và ROWS.
- Mệnh đề FROM xác định nguồn dữ liệu đa chiều (khối) nào được sử dụng để trích lọc dữ liệu vào tập kết quả của phát biểu SELECT. Trong ví dụ là từ khối dữ liệu **[QL Diem DW]**

---

---

---

---

---

---

---

### Giải thích ví dụ 1 (2)

- Mệnh đề WHERE nếu có, dùng để xác định chiều cắt của khối dữ liệu, nhằm giới hạn tập kết quả theo các chiều thành viên. Trong ví dụ, dữ liệu được cắt lát theo thành viên **[Nam Hoc].[2004]** của chiều **[Dim Thoi Gian]**.

---

---

---

---

---

---

---

### *Sự khác biệt giữa ngôn ngữ SQL và ngôn ngữ MDX*

- Cú pháp của câu truy vấn MDX tương tự như cú pháp SQL, tuy nhiên có một số
  - Cú pháp MDX phân biệt các tập hợp bằng việc dùng dấu ngoặc nhọn {} bao quanh các bộ hay các thành viên khác biệt sau:

---

---

---

---

---

---

---

### Sự khác biệt giữa ngôn ngữ SQL và ngôn ngữ MDX (2)

- Câu truy vấn MDX có thể chứa tới 128 chiều nhưng chỉ có 5 chiều đầu tiên là có aliases (tên phụ).
- Trong câu truy vấn SQL, mệnh đề FROM có thể chỉ đến nhiều bảng dữ liệu.
- Tuy nhiên, mệnh đề FROM trong câu truy vấn MDX bị giới hạn bởi một khối duy nhất.

---

---

---

---

---

---

---

---

### Sự khác biệt giữa ngôn ngữ SQL và ngôn ngữ MDX (3)

- Mệnh đề WHERE được dùng để mô tả chiều cắt dữ liệu.
- Nếu một chiều không được đề cập trong mệnh đề WHERE, SQL Analysis Services vẫn xem nó như một chiều cắt, nhưng được lọc theo số thành viên (Members) mặc định của nó.
- Mệnh đề WHERE có thể thay đổi tiến trình lọc cho một chiều nào đó, giúp tinh chỉnh dữ liệu kết quả.

---

---

---

---

---

---

---

---

### Ta xét ví dụ 2:

```
SELECT [Measures].[Fact Diem Tot Nghiep Count] on COLUMNS ,
       [Dim Xep Loai TN].[Xep Loai TN].Members on ROWS
FROM   [QL Diem DW]
WHERE  [Dim Thoi Gian].[Nam Hoc].[2004]
```

### Ta có kết quả:

	Fact Diem Tot Nghiep Count
All	376
G	14
K	177
TB	16
TBK	169
XS	[null]

---

---

---

---

---

---

---

---

### Ta xét ví dụ 3:

```
SELECT [Dim Don Vi].[Bo Mon].members on columns,
       Dim Xep Loai TN].[Xep Loai TN].members on ROWS
FROM   [QL Diem DW]
WHERE  [Dim Thoi Gian].[Nam Hoc].[2004]
```

### Ta có kết quả:

	All	HTTTQL	KTDN	KTH	KTKT	LUAT	QTKD	TCNH
All	376	14	100	107	78	13	(null)	64
G	14	(null)	7	2	4	(null)	(null)	1
K	177	7	57	40	35	4	(null)	34
TB	16	(null)	4	10	(null)	(null)	(null)	2
TBK	169	7	32	55	39	9	(null)	27
XS	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)

---

---

---

---

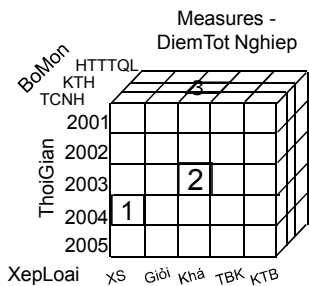
---

---

---

---

- 1: (BoMon.TCINH, XepLoai.XS, ThoiGian.[2004])
- 2: (BoMon.TCINH, XepLoai.Kha, ThoiGian.[2003])
- 3: (BoMon.KTH, XepLoai.Kha, ThoiGian.[2001])



Tất cả các ô đều có một cái tên và giá trị tương ứng với các chiều

---

---

---

---

---

---

---

---

### Một số hàm trong MDX Query

---

---

---

---

---

---

---

---



## Các hàm thông dụng trong MDX

- CrossJoin() – Tích của các thành viên hay các bộ trong hai tập hợp khác nhau
- Ví dụ: CrossJoin hai chiều DonVi và XepLoai

```
SELECT CrossJoin([Dim Don Vi].[Bo Mon].CHILDREN,[Dim
Xep Loai TN].[Xep Loai TN].CHILDREN) on
AXIS(0),
[Dim Thời Gian].[Nam Hoc].members on AXIS(1)
FROM [QL Diem DW]
WHERE [Measures].[Fact Diem Tot Nghiep Count]
```

Kết quả

	HTTQL	HTTQL	HTTQL	HTTQL	HTTQL	KTON	KTON	KTON	KTON	KTON	KTH	KTH	KTH	KTH
	G	K	18	18	25	G	K	18	25	G	K	18	25	
2001	(null)	25	(null)	44	(null)	57	363	13	240	(null)	16	225	48	284
2002	(null)	(null)	(null)	(null)	(null)	2	13	(null)	8	(null)	2	12	4	22
2003	(null)	(null)	(null)	(null)	(null)	4	26	3	12	(null)	1	14	10	24
2004	(null)	(null)	(null)	(null)	(null)	1	42	2	30	(null)	2	24	8	46
2005	(null)	7	(null)	7	(null)	7	67	4	32	(null)	2	40	10	50
2006	(null)	6	(null)	4	(null)	3	46	2	27	(null)	4	33	6	58
2007	(null)	7	(null)	5	(null)	6	62	1	44	(null)	(null)	31	3	64
2008	(null)	11	(null)	17	(null)	12	68	(null)	55	(null)	2	45	2	59
2009	2	4	(null)	11	(null)	2	49	1	32	(null)	3	26	5	55

## Các hàm thông dụng trong MDX (2)

- Filter() – thu gọn một tập hợp bằng cách chỉ lấy những phần tử thỏa mãn các tiêu chí nào đó
  - Đối số
    - Tập hợp
    - Biểu thức luận lý
  - Trả về tập con
  - Ví dụ: Filter (
 

```
{ [DimDonVi].[BoMon].Members },
[Measures].[DiemTotNghiep].[Kha] >= 20
}
```

## Các hàm thông dụng trong MDX (3)

- Order() – sắp thứ tự một tập hợp
  - Đối số
    - Tập hợp
    - Tiêu chí sắp thứ tự
    - Tùy chọn cờ (ASC, DESC,...)
  - Ví dụ: Order (
 

```
{ [DimDonVi].[BoMon].Members },
([Measures].[DiemTotNghiep],
[ThờiGian].[Nam].[2000]),
DESC
}
```

## Bài tập mở rộng

- Sinh viên tìm hiểu thêm các hàm sau:
  - TopCount
  - Order
  - ParallelPeriod
  - Sum
  - Strtomember
  - iif
  - Lag
  - ...

---

---

---

---

---

---

---