

# Lập Trình PL/SQL

## Data Types



BATIZENS

### Người trình bày:



Đoàn Thị Kim Nhung



ITBA Team

# Mục Lục

01

Nhóm Data Types  
Data Types Group

02

Chi tiết nhóm Data Types  
Data Types Group Detail



BATIZENS



# Từ ngữ viết tắt và định nghĩa

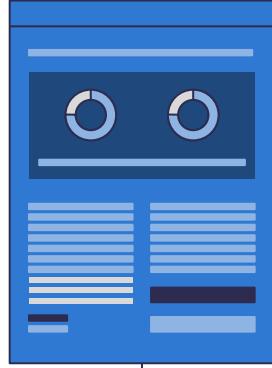
#	Từ ngữ viết tắt	Định nghĩa
1	DT	Data Types
2	DL	Dữ liệu
3	p	Precision – tổng số chữ số mà số có thể có (gồm cả chữ số trước và sau dấu thập phân)
4	s	Scale – số chữ số có thể có sau dấu thập phân
5	signed	Có dấu – số nguyên là số dương, số âm hoặc số 0
6	unsigned	Không dấu – số nguyên là số dương hoặc số 0
7	Z	Zahlen – số nguyên
8	R	Real – số thực
9	CSDL	Cơ sở dữ liệu





01

# Nhóm Data Types





# Nhóm Data Types

## Loại vô hướng (Scalar Types)

### Số (Numeric)

BINARY\_INTEGER  
NUMBER  
NATURAL  
NATURALNL  
POSITIVE  
POSITIVEN  
SIGNTYPE  
  
PLS\_INTEGER  
NUMBER  
INT/INTEGER  
SMALLINT  
DEC/DECIMAL  
NUMERIC  
REAL  
DOUBLE PRECISION  
FLOAT

PLS\_INTEGER  
SIMPLE\_INTEGER  
NATURAL  
NATURALNL  
POSITIVE  
POSITIVEN  
SIGNTYPE

### Logic (Boolean)

BOOLEAN  
BOOLEAN

### Ký tự (Character)

CHARACTER  
CHAR  
NCHAR  
VARCHAR  
VARCHAR2  
STRING  
NVARCHAR2  
LONG  
LONG RAW  
RAW  
ROWID  
UROWID

### Ngày giờ (Datetime)

DATETIME  
DATE  
TIMESTAMP  
TIMESTAMP WITH TIME ZONE  
TIMESTAMP WITH LOCAL TIME ZONE  
INTERVAL YEAR TO MONTH  
INTERVAL DAY TO SECOND

## Loại đối tượng lớn (LOB Types)

BFILE  
BLOB  
CLOB  
NCLOB

## Loại hỗn hợp (Composite Types)

ARRAY  
RECORD  
TABLE

## Loại tham chiếu (Reference Types)

REF CURSOR  
REF object\_type





03

# Chi tiết nhóm Data Types





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## Loại vô hướng (Scalar Types)

1. Số (Numeric)

Các giá trị số, các phép tính số học được thực hiện.

2. Ký tự (Character)

Các giá trị chữ và số, đại diện cho các ký tự đơn hoặc chuỗi ký tự.

3. Logic (Boolean)

Các giá trị logic, các phép toán logic được thực hiện.

4. Ngày giờ (Datetime)

Ngày và giờ.





DT

# 1. Loại vô hướng (Scalar Types)



## 1. Số (Numeric):

Scalar Types

Composite Types

Reference Types

LOB Types

### Số (Numeric)





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 1. Số (Numeric):

Data Type	Phạm vi	Bộ nhớ	Mô tả
NUMBER			
NUMBER(p,s)/NUMBER - p: [1; 38] - s: [-84; 127]	<ul style="list-style-type: none"> <li>- Range: <math>[-10^{130} + 1; 10^{130} - 1]</math></li> <li>- Default: NUMBER (38,0)</li> <li>- Variable-length</li> <li>- NULL</li> </ul>	22 bytes	<ul style="list-style-type: none"> <li>- Lưu trữ giá trị số nguyên và số thực.</li> <li>- Độ chính xác và kích thước tùy chọn.</li> <li>- Scale âm: Trong các giá trị <math>&lt; 1</math>.</li> <li>- Scale dương: Trong các giá trị <math>\geq 1</math>.</li> <li>- Hiệu suất cao.</li> </ul>
INT/INTEGER	<ul style="list-style-type: none"> <li>- INT: [1; 38]</li> <li>- Range: <math>[-10^{38}+1; 10^{38}-1]</math></li> <li>- Default: INT (38)</li> <li>- Variable-length</li> <li>- NULL</li> </ul>	22 bytes	<ul style="list-style-type: none"> <li>- Lưu trữ giá trị số nguyên.</li> <li>- Có cùng phạm vi và đặc điểm với NUMBER(38,0).</li> </ul>
SMALLINT	<ul style="list-style-type: none"> <li>- SMALLINT: [1; 38]</li> <li>- Range: <math>[-10^{38}+1; 10^{38}-1]</math></li> <li>- Default: SMALLINT (38)</li> <li>- Variable-length</li> <li>- NULL</li> </ul>	22 bytes	<ul style="list-style-type: none"> <li>- Lưu trữ giá trị số nguyên.</li> <li>- Trong phạm vi rộng như NUMBER(38,0), thường được sử dụng với phạm vi từ -32,768 đến 32,767.</li> </ul>





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 1. Số (Numeric):

Data Type	Phạm vi	Bộ nhớ	Mô tả
NUMBER			
DEC/DECIMAL(p,s) - p: [1; 38] - s: [0; p]	<ul style="list-style-type: none"> <li>- Range: [-10^38 + 1; 10^38 - 1]</li> <li>- Default: DEC (38,0)</li> <li>- Variable-length</li> <li>- NULL</li> </ul>	22 bytes	<ul style="list-style-type: none"> <li>- Lưu trữ giá trị số nguyên và số thực.</li> <li>- Độ chính xác cố định.</li> <li>- s &lt;= p.</li> <li>- Độ chính xác cao nhất.</li> </ul>
NUMERIC (p,s) - p: [1; 38] - s: [0; p]	<ul style="list-style-type: none"> <li>- Range: [-10^38 + 1; 10^38 - 1]</li> <li>- Default: NUMERIC (38,0)</li> <li>- Variable-length</li> <li>- NULL</li> </ul>	22 bytes	<ul style="list-style-type: none"> <li>- Lưu trữ giá trị số nguyên và số thực.</li> <li>- s &lt;= p.</li> <li>- Độ chính xác cao, hiệu suất cao.</li> </ul>
REAL	<ul style="list-style-type: none"> <li>- Range: [-3.4E38; 3.4E38]</li> <li>- Variable-length</li> <li>- NULL</li> </ul>	4 bytes	<ul style="list-style-type: none"> <li>- Lưu trữ giá trị số thực.</li> <li>- Sử dụng để đại diện cho các giá trị số thực trong phạm vi đặc biệt, nhưng không cần độ chính xác cao như kiểu DOUBLE PRECISION.</li> </ul>





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 1. Số (Numeric):

Data Type	Phạm vi	Bộ nhớ	Mô tả
NUMBER			
DOUBLE PRECISION	<ul style="list-style-type: none"> <li>- Range: [-1.0E-308; 1.0E+308]</li> <li>- Variable-length</li> <li>- NULL</li> </ul>	8 bytes	<ul style="list-style-type: none"> <li>- Lưu trữ giá trị số thực.</li> <li>- Có phạm vi và độ chính xác cao hơn so với kiểu REAL hoặc FLOAT.</li> </ul>
FLOAT (n) - n: [1; 38]	<ul style="list-style-type: none"> <li>- Range: [-1.0E-126; 1.0E+126]</li> <li>- Default: FLOAT (38)</li> <li>- NULL</li> </ul>	4/8 bytes	<ul style="list-style-type: none"> <li>- Lưu trữ giá trị số thực.</li> </ul>
BINARY_INTEGER			
BINARY_INTEGER	<ul style="list-style-type: none"> <li>- Range: [-2,147,483,648; 2,147,483,647]</li> <li>- NULL</li> </ul>	4 bytes	<ul style="list-style-type: none"> <li>- Lưu trữ các số nguyên có dấu.</li> <li>- Sử dụng ít bộ nhớ hơn NUMBER.</li> <li>- Hiệu suất thấp hơn PLS_INTEGER.</li> </ul>
NATURAL	<ul style="list-style-type: none"> <li>- Range: [0; 2,147,483,647]</li> <li>- NULL</li> </ul>	4 bytes	Lưu trữ số nguyên không âm (từ 0 trở lên).





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 1. Số (Numeric):

Data Type	Phạm vi	Bộ nhớ	Mô tả
BINARY_INTEGER			
NATURALNL	<ul style="list-style-type: none"> <li>- Range: [0; 2,147,483,647]</li> <li>- NOT NULL</li> </ul>	4 bytes	Lưu trữ số nguyên không âm (từ 0 trở lên).
POSITIVE	<ul style="list-style-type: none"> <li>- Range: [1; 2,147,483,647]</li> <li>- NULL</li> </ul>	4 bytes	Lưu trữ số nguyên dương (từ 1 trở lên).
POSITIVEN	<ul style="list-style-type: none"> <li>- Range: [1; 2,147,483,647]</li> <li>- NOT NULL</li> </ul>	4 bytes	Lưu trữ số nguyên dương (từ 1 trở lên).
SIGNTYPE	<ul style="list-style-type: none"> <li>- Range: [-1; 1] = {-1; 0; 1}</li> <li>- NULL</li> </ul>	4 bytes	Lưu trữ các giá trị -1, 0 hoặc 1.





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 1. Số (Numeric):

Data Type	Phạm vi	Bộ nhớ	Mô tả
PLS_INTEGER			
PLS_INTEGER	<ul style="list-style-type: none"> <li>- Range: [-2,147,483,648; 2,147,483,647]</li> <li>- NULL</li> </ul>	4 bytes	<ul style="list-style-type: none"> <li>- Lưu trữ các số nguyên có dấu.</li> <li>- Hiệu suất cao hơn BINARY_INTEGER và NUMBER.</li> </ul>
SIMPLE_INTEGER	<ul style="list-style-type: none"> <li>- Range: [-2,147,483,648; 2,147,483,647]</li> <li>- NOT NULL</li> </ul>	4 bytes	<ul style="list-style-type: none"> <li>- Lưu trữ các số nguyên có dấu.</li> <li>- Hiệu suất cao hơn BINARY_INTEGER và NUMBER.</li> </ul>
NATURAL	<ul style="list-style-type: none"> <li>- Range: [0; 2,147,483,647]</li> <li>- NULL</li> </ul>	4 bytes	Lưu trữ số nguyên không âm (từ 0 trở lên).
NATURALNL	<ul style="list-style-type: none"> <li>- Range: [0; 2,147,483,647]</li> <li>- NOT NULL</li> </ul>	4 bytes	Lưu trữ số nguyên không âm (từ 0 trở lên).
POSITIVE	<ul style="list-style-type: none"> <li>- Range: [1; 2,147,483,647]</li> <li>- NULL</li> </ul>	4 bytes	Lưu trữ số nguyên dương (từ 1 trở lên).





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 1. Số (Numeric):

Data Type	Phạm vi	Bộ nhớ	Mô tả
PLS_INTEGER			
POSITIVEN	- Range: [1; 2,147,483,647] - NOT NULL	4 bytes	Lưu trữ số nguyên dương (từ 1 trở lên).
SIGNTYPE	- Range: [-1; 1] = {-1; 0; 1} - NULL	4 bytes	Lưu trữ các giá trị -1, 0 hoặc 1.





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite  
TypesReference  
Types

LOB Types

## 2. Ký tự (Character):

**Ký tự (Character)**

{ CHARACTER }

1. CHARACTER	{	STRING
2. CHAR	{	STRING
3. NCHAR	{	STRING
4. VARCHAR	{	STRING
5. VARCHAR2	{	STRING
6. STRING	{	STRING
7. NVARCHAR2	{	STRING
8. LONG	{	STRING
9. LONG RAW	{	BINARY
10. RAW	{	BINARY
11. ROWID	{	STRING
12. UROWID	{	STRING

Các giá trị là chuỗi ký tự gồm chữ cái (ALPHA), chữ số (NUMBER) và các ký tự đặc biệt (SPECIAL).





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 2. Ký tự (Character):

Data Type	Phạm vi	Bộ nhớ	Mô tả
CHARACTER			
CHARACTER (n) - n: [1; 2,000]	- Fixed-length - NULL	Thay đổi	Lưu trữ chuỗi ký tự với độ dài cố định.
CHAR (n) - n: [1; 2,000]	- Fixed-length - NULL	Thay đổi	Lưu trữ chuỗi ký tự với độ dài cố định.
NCHAR (n) - n: [1; 2,000]	- Fixed-length - National character type. - NULL	Thay đổi	Lưu trữ chuỗi ký tự có độ dài cố định sử dụng bộ mã hóa ký tự quốc tế Unicode – các ký tự từ nhiều ngôn ngữ khác nhau trên thế giới.
VARCHAR (n) - n: [1; 4,000]	- Variable-length - NULL	Thay đổi	Lưu trữ chuỗi ký tự có độ dài biến đổi – giúp tiết kiệm bộ nhớ khi lưu trữ các chuỗi ký tự có độ dài không đồng đều.





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 2. Ký tự (Character):

Data Type	Phạm vi	Bộ nhớ	Mô tả
CHARACTER			
VARCHAR2 (n) - n: [1; 4,000]	- Variable-length - NULL	Thay đổi	<ul style="list-style-type: none"> <li>Lưu trữ chuỗi ký tự có độ dài biến đổi.</li> <li>Hoạt động theo cùng một cách trong tất cả các phiên bản tương lai.</li> </ul>
NVARCHAR2 (n) - n: [1; 4000]	- Variable-length - National character type. - NULL	Thay đổi	<ul style="list-style-type: none"> <li>Lưu trữ chuỗi ký tự có độ dài biến đổi sử dụng bộ mã hóa ký tự quốc tế Unicode – các ký tự từ nhiều ngôn ngữ khác nhau trên thế giới.</li> <li>Hoạt động theo cùng một cách trong tất cả các phiên bản tương lai.</li> </ul>





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 2. Ký tự (Character):

Data Type	Phạm vi	Bộ nhớ	Mô tả
CHARACTER			
LONG	<ul style="list-style-type: none"> <li>- LONG: [1; ~ 2 tỷ] ~ 2GB</li> <li>- Variable-length ~ CLOB</li> <li>- String</li> <li>- NULL</li> </ul>	2 gigabytes	<ul style="list-style-type: none"> <li>- Lưu trữ chuỗi ký tự có độ dài biến đổi lên đến 2GB.</li> <li>- Một kiểu dữ liệu lỗi thời, có nhiều hạn chế và bị thay thế bởi các kiểu dữ liệu mới hơn như CLOB.</li> </ul>
LONG RAW	<ul style="list-style-type: none"> <li>- LONG RAW: [1; ~ 2GB]</li> <li>- Variable-length ~ BLOB</li> <li>- Image, audio, video, file,...</li> <li>- NULL</li> </ul>	2 gigabytes	<ul style="list-style-type: none"> <li>- Lưu trữ dữ liệu nhị phân (hình ảnh, âm thanh, video, tập tin văn bản) lớn với độ dài biến đổi lên đến 2GB.</li> <li>- Một kiểu dữ liệu lỗi thời, có nhiều hạn chế và bị thay thế bởi các kiểu dữ liệu mới hơn như BLOB.</li> </ul>
RAW	<ul style="list-style-type: none"> <li>- RAW: [1; ~ 2000B]</li> <li>- Fixed-length / Variable-length</li> <li>- Image, audio, video, file,...</li> <li>- NULL</li> </ul>		Lưu trữ dữ liệu nhị phân.





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 2. Ký tự (Character):

Data Type	Phạm vi	Bộ nhớ	Mô tả
CHARACTER			
ROWID	<ul style="list-style-type: none"> <li>- ROWID: [18] = OOOOOFFFBBBBBBRRR</li> <li>- Fixed-length</li> <li>- Old data type</li> <li>- NULL</li> </ul>		<ul style="list-style-type: none"> <li>- Kiểu dữ liệu đặc biệt trong Oracle Database, được sử dụng để lưu trữ địa chỉ thực tế của một hàng trong một bảng. Nó là một giá trị duy nhất cho mỗi hàng trong một bảng và có thể được sử dụng để nhanh chóng xác định và truy cập hàng đó.</li> <li>- ROWID thường được sử dụng với các bảng truyền thống.</li> </ul>
UROWID	<ul style="list-style-type: none"> <li>- UROWID: [18] = OOOOOFFFBBBBBBRRR</li> <li>- Fixed-length</li> <li>- New data type</li> <li>- NULL</li> </ul>		<ul style="list-style-type: none"> <li>- Lưu trữ nhiều loại ROWID khác nhau, bao gồm ROWID của các bảng phân vùng, bảng tổ chức theo chỉ mục (IOT), và bảng đối tượng.</li> <li>- UROWID linh hoạt hơn và có thể được sử dụng với các cấu trúc lưu trữ đặc biệt như IOT.</li> </ul>





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 3. Logic (Boolean):

**Logic (Boolean)**

{ BOOLEAN { 1. BOOLEAN }

Biểu diễn giá trị đúng (TRUE) hoặc sai (FALSE) hoặc NULL. Thực hiện các phép toán logic và điều kiện.

Data Type	Phạm vi	Bộ nhớ	Mô tả
BOOLEAN			
BOOLEAN	FALSE; TRUE; NULL	Thay đổi	- Biểu diễn giá trị logic TRUE, FALSE hoặc NULL. - Thực hiện các phép toán logic và điều kiện.





DT

# 1. Loại vô hướng (Scalar Types)



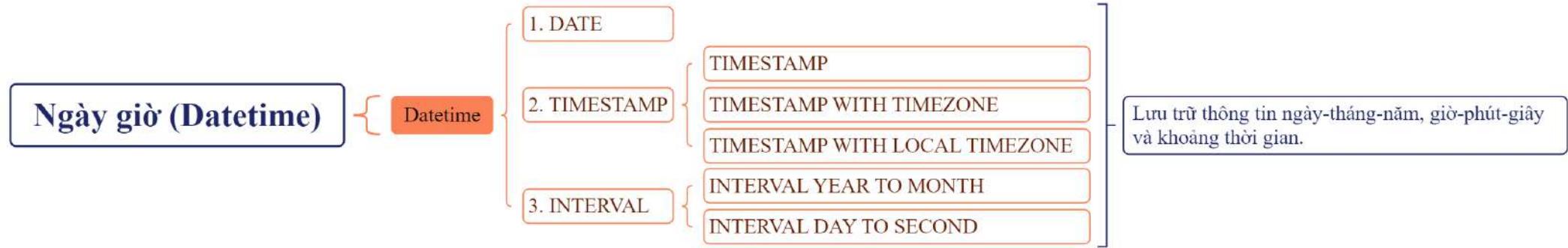
Scalar Types

Composite Types

Reference Types

LOB Types

## 4. Ngày giờ (Datetime):



Data Type	Phạm vi	Bộ nhớ	Mô tả
DATETIME			
DATE	<ul style="list-style-type: none"> <li>- Range: [01/01/4712; 31/12/9999]</li> <li>- Format: DD-MON-YY (Fixed-length)</li> <li>- SYSDATE: The current date and time</li> <li>- Default: NLS_DATE_FORMAT</li> <li>- NULL</li> </ul>		Lưu trữ giá trị ngày tháng năm. Nó không bao gồm thông tin về giờ, phút hoặc giây.





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 4. Ngày giờ (Datetime):

Data Type	Phạm vi	Bộ nhớ	Mô tả
<b>DATETIME</b>			
TIMESTAMP (p) - p: [0; 9]	<ul style="list-style-type: none"> <li>- Range: [01/01/4712; 31/12/9999]</li> <li>- Format: YYYY-MM-DD HH24:MI:SS.FF</li> <li>- SYSTIMESTAMP: The current timestamp</li> <li>- Default: TIMESTAMP (6), NLS_TIMESTAMP_FORMAT</li> </ul>		<ul style="list-style-type: none"> <li>- Lưu trữ thông tin về ngày, tháng, năm và thời gian trong ngày (giờ, phút, giây, và đôi khi là phần nghìn giây).</li> <li>- NULL.</li> </ul>
TIMESTAMP (fractional seconds precision - p) WITH TIME ZONE - p: [0; 9]	<ul style="list-style-type: none"> <li>- Range: [01/01/4712; 31/12/9999]</li> <li>- Format: YYYY-MM-DD HH24:MI:SS.FF TZH:TZM</li> <li>- SYSTIMESTAMP: The current timestamp</li> <li>- Default: TIMESTAMP (6) WITH TIME ZONE, NLS_TIMESTAMP_TZ_FORMAT</li> </ul>		<ul style="list-style-type: none"> <li>- Lưu trữ thông tin về ngày và thời gian cùng với múi giờ. Tính chất đặc biệt của TIMESTAMP WITH TIME ZONE là nó bao gồm cả thông tin về múi giờ, điều này cho phép biểu diễn các thời điểm đồng thời ở các vị trí khác nhau trên thế giới.</li> <li>- NULL.</li> </ul>





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 4. Ngày giờ (Datetime):

Data Type	Phạm vi	Bộ nhớ	Mô tả
DATETIME			
TIMESTAMP (fractional seconds precision - p) WITH LOCAL TIME ZONE - p: [0; 9]	<ul style="list-style-type: none"> <li>- Range: [01/01/4712; 31/12/9999]</li> <li>- Format: YYYY-MM-DD HH24:MI:SS.FF (Fixed-length)</li> <li>- SYSTIMESTAMP: The current timestamp</li> <li>- Default: TIMESTAMP (6) WITH LOCAL TIME ZONE, NLS_TIMESTAMP_FORMAT</li> <li>- NULL</li> </ul>		Lưu trữ thông tin về ngày và thời gian, nhưng dựa trên múi giờ của hệ thống cục bộ.
INTERVAL YEAR (year precision - p) TO MONTH - p: [0; 4]	<ul style="list-style-type: none"> <li>- YEAR: [0; 999,999,999]</li> <li>- MONTH: [0; 11]</li> <li>- Default: INTERVAL YEAR (2) TO MONTH</li> </ul>		Lưu trữ một khoảng thời gian từ một số năm đến một số tháng, không bao gồm ngày hoặc thời gian chi tiết.





DT

# 1. Loại vô hướng (Scalar Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## 4. Ngày giờ (Datetime):

Data Type	Phạm vi	Bộ nhớ	Mô tả
DATETIME			
INTERVAL DAY (day precision - p1) TO SECOND (fractional seconds precision - p2) - p1: [0; 9] - p2: [0; 9]	- DAY: [0; 999,999,999] - SECOND: [0; 59.999,999,999] - Default: INTERVAL DAY (2) TO SECOND (6)		Lưu trữ một khoảng thời gian dưới dạng số ngày, giờ, phút, giây và có thể bao gồm cả phần thập phân của giây.





DT

## 2. Loại hỗn hợp (Composite Types)



Scalar Types

Composite Types

Reference Types

LOB Types

### Loại hỗn hợp (Composite Types)

#### 1. Mảng (Array)

Kiểu dữ liệu mảng thường được tham chiếu đến như là VARRAY (Variable-Size Array) hoặc Nested Table.

#### 2. Bản ghi (Record)

Một cấu trúc dữ liệu tùy chỉnh, chứa nhiều trường dữ liệu khác nhau có thể là các kiểu dữ liệu cơ bản như INTEGER, VARCHAR2, DATE, hoặc thậm chí là các kiểu dữ liệu phức tạp như VARRAY hoặc Nested Table.

#### 3. Bảng (Table)

Định nghĩa một cấu trúc dữ liệu tương tự như bảng (tạm thời) trong cơ sở dữ liệu, cho phép lưu trữ nhiều giá trị trong một biến hoặc một cột dữ liệu.





DT

## 2. Loại hỗn hợp (Composite Types)



Scalar Types

Composite Types

Reference Types

LOB Types

### 1. Mảng (Array):

Data Type	Phạm vi	Bộ nhớ	Mô tả
ARRAY			
VARRAY	<ul style="list-style-type: none"> <li>- n: [1; 2,147,483,647 ]</li> <li>- Fixed-length</li> </ul>		<ul style="list-style-type: none"> <li>- Lưu trữ một tập hợp các phần tử có kích thước thay đổi nhưng với giới hạn tối đa cố định.</li> <li>- Nó hữu ích khi làm việc với các tập hợp nhỏ có kích thước được xác định trước.</li> </ul>
IS TABLE OF (Nested Table)	Variable-length		<ul style="list-style-type: none"> <li>- Một loại cấu trúc dữ liệu được sử dụng để lưu trữ một tập hợp các phần tử có kích thước thay đổi, tương tự như một mảng nhưng không có giới hạn cố định về số lượng phần tử. Nested table có thể được lưu trữ trong CSDL.</li> <li>- Các phần tử của nó có thể được lưu trữ không liên tiếp trong cơ sở dữ liệu và có thể mở rộng hoặc rút gọn một cách linh hoạt.</li> </ul>





DT

## 2. Loại hỗn hợp (Composite Types)



Scalar Types

Composite Types

Reference Types

LOB Types

### 2. Bản ghi (Record):

Data Type	Phạm vi	Bộ nhớ	Mô tả
RECORD			
	IS RECORD		<ul style="list-style-type: none"><li>- Một loại cấu trúc dữ liệu phức hợp cho phép nhóm các trường dữ liệu không đồng nhất vào một đơn vị logic duy nhất.</li><li>- RECORD rất hữu ích khi cần làm việc với nhiều loại dữ liệu khác nhau mà có liên quan với nhau, chẳng hạn như các cột của một hàng trong bảng CSDL.</li><li>- RECORD có thể lồng nhau, nghĩa là một trường của RECORD có thể là một RECORD khác.</li></ul>





DT

## 2. Loại hỗn hợp (Composite Types)



Scalar Types

Composite Types

Reference Types

LOB Types

### 3. Bảng (Table):

Data Type	Phạm vi	Bộ nhớ	Mô tả
TABLE			
IS TABLE OF	Variable-length		<ul style="list-style-type: none"><li>- Kiểu dữ liệu phức hợp cho phép lưu trữ và làm việc với các tập hợp các phần tử, tương tự như một mảng (array).</li><li>- Lưu trữ tạm thời các tập hợp các giá trị mà cần thao tác trong một khối PL/SQL.</li></ul>





DT

### 3. Loại tham chiếu (Reference Types)



Scalar Types

Composite Types

Reference Types

LOB Types



Scalar Types

Composite Types

Reference Types

LOB Types

#### Loại tham chiếu (Reference Types)

##### 1. REF CURSOR

- Một loại biến được sử dụng để tham chiếu tới kết quả của một câu lệnh SELECT trong CSDL.
- Nó giống như một con trỏ hoặc một tham chiếu đến một tập hợp các dòng dữ liệu được trả về từ một truy vấn SQL.
- REF CURSOR cho phép truy cập dữ liệu từ kết quả của truy vấn một cách linh hoạt trong các chương trình PL/SQL.

##### 2. REF object\_type

- Trong PL/SQL, REF là một kiểu dữ liệu được sử dụng để tham chiếu đến một đối tượng (object) trong cơ sở dữ liệu. Đối tượng này có thể là một đối tượng cấp cao (object type) đã được định nghĩa trước trong CSDL.
- Khi sử dụng REF, có thể tạo một tham chiếu tới một đối tượng cụ thể trong CSDL mà không cần lấy toàn bộ DL của đối tượng đó. Điều này giúp tối ưu hóa hiệu suất và tiết kiệm tài nguyên.





DT

## 4. Loại đối tượng lớn (LOB Types)



Scalar Types

Composite Types

Reference Types

LOB Types

### Loại đối tượng lớn (LOB Types)

1. BFILE (Binary File)

Lưu trữ các liên kết đến các tập tin nhị phân lớn nằm ngoài cơ sở dữ liệu.

2. BLOB (Binary Large Object)

Lưu trữ dữ liệu nhị phân lớn như hình ảnh, video hoặc âm thanh.

3. CLOB (Character Large Object)

Lưu trữ dữ liệu văn bản dài hoặc chuỗi ký tự lớn.

4. NCLOB (National Character Large Object)

Lưu trữ dữ liệu văn bản dài hoặc chuỗi ký tự lớn có mã ký tự quốc tế (Unicode - các ký tự từ nhiều ngôn ngữ khác nhau trên thế giới).





DT

# 4. Loại đối tượng lớn (LOB Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## Loại đối tượng lớn (LOB Types)

Data Type	Phạm vi	Bộ nhớ	Mô tả
LOB TYPES			
BFILE	<ul style="list-style-type: none"><li>- BFILE: [1; ~4000B]</li><li>- Lưu trữ tham chiếu đến các tệp nhị phân bên ngoài CSDL.</li></ul>		Lưu trữ các liên kết đến các tập tin nhị phân lớn nằm ngoài cơ sở dữ liệu.
BLOB	<ul style="list-style-type: none"><li>- BLOB: [1; ~ 4GB]</li><li>- Variable-length</li><li>- Image, audio, video, file,...</li><li>- NULL</li></ul>		Lưu trữ dữ liệu nhị phân lớn như hình ảnh, video hoặc âm thanh.
CLOB	<ul style="list-style-type: none"><li>- CLOB: [1; ~ 4GB]</li><li>- Variable-length</li><li>- String</li><li>- NULL</li></ul>		Lưu trữ dữ liệu văn bản dài hoặc chuỗi ký tự lớn.





DT

# 4. Loại đối tượng lớn (LOB Types)



Scalar Types

Composite Types

Reference Types

LOB Types

## Loại đối tượng lớn (LOB Types)

Data Type	Phạm vi	Bộ nhớ	Mô tả
LOB TYPES			
NCLOB	<ul style="list-style-type: none"><li>- NCLOB: [1; ~ 4GB]</li><li>- Variable-length</li><li>- National character type.</li><li>- String</li><li>- NULL</li></ul>		<ul style="list-style-type: none"><li>- Lưu trữ dữ liệu văn bản dài hoặc chuỗi ký tự lớn có mã ký tự quốc tế (Unicode).</li><li>- Khi cần lưu trữ dữ liệu văn bản chứa các ký tự không phải ASCII hoặc cần hỗ trợ nhiều ngôn ngữ khác nhau, NCLOB là lựa chọn phù hợp hơn.</li></ul>





B A T I Z E N S



*Trân trọng cảm ơn!*

