

## # ORACLE MASTERの基礎が学べる本

ノートブック: 210\_メモ

作成: 2016/02/08 23:40

更新: 2016/02/10 7:35

作成者: Tsutsui tomoaki

URL: file:///D:/tools/kobito/resources/app/versions/1.5.0.asar//index-atom-s...

---

# # ORACLE MASTERの基礎が学べる本

## 一日目 \* データベースの利用方法

---

### OPAC(ONLINE PUBLIC ACCESS CATALOG)

図書目録カードを電子化しONLINEで検索可能にしたシステム

## 二日目 \* 学習の準備

---

### ループバックアダプターについて

DHCPが有効の場合はループバックアダプターをインストールする必要がある。

#### 【手順】

- ①ファイル名を指定して実行  
「hdwwiz」と入力
- ②ネットワークアダプターを選択
- ③「Microsoft Loopback Adapter」を選択
- ④ローカルエリア接続2に作成されるので、IPアドレスを  
IPアドレス：10.10.10.10  
サブネット：255.255.255.0
- ⑤HOSTSを設定する  
「C:¥Windows¥System32¥drivers¥etc¥hosts」  
を開いて、最終行に  
「10.10.10.10, コンピュータ名」  
と入力し、上書き保存する

## ■リスナーの作成

Net Configuration Assistantというツールを使用する

リスナー名：LISTENER

標準ポート：1521

### ○コマンド

「 LSNRCTL 」を使用します

\* 構文 リスナーの起動

LSNRCTL start

\* 構文 リスナーの停止

LSNRCTL stop

\* 構文 リスナーの状態表示

LSNRCTL status

## データベースを作成

Database Configuration Assistant(DBCA)というツールを使用

## 二日目 \* ツールの使い方

---

## Oracle Enterprise Managerを使ってみよう

### 基本操作

\* アカウント

sys データベースの管理者アカウント。高度な管理作業を行うときに使用

system データベースの管理者アカウント。通常の管理作業

scott サンプルのユーザーアカウント。基底ではアカウントがロックされている(使用できない状態)

【手順】

「 <https://localhost:1158/em> 」

と入力して接続します。(ポートはさっき設定したやつ)

\* 接続できない場合

- ・ Loopbackアダプターが構成されているかどうか
- ・ リスナーが開始されているかどうか
- ・ dbconsoleプロセスが起動しているかどうか

\* ポートの確認

```
SELECT DBMS_XDB_CONFIG.GETHTTPSPORT() FROM DUAL;
```

\* ポートの設定

```
EXEC DBMS_XDB_CONFIG.SETHTTPSPORT(5500);
```

\* dbconsoleプロセスの起動

Enterprise Managerを使用するためには

リスナー

dbconsoleプロセス

を起動していなければなりません。

dbconsoleプロセスを開始したり状態を確認するには

EMCTLを使用します。

EMCTLを使用するにはORACLE\_SID環境変数を設定しておく必要があります。

\* ORACLEサービス

Oracle Database : OracleServiceORCL

リスナー : OracleOraDB12c\_home1TNSListener

dbconsole : OracleDBConsoleorcl

## SQL\*Plusを使ってみよう

○SQL文の保存

```
select objectname from user_objects
```

```
save d:¥work¥010作業中¥oracle¥lesson.sql
```

```
start d:¥work¥010作業中¥oracle¥lesson.sql
```

```
d:¥work¥010作業中¥oracle>sqlplus scott/tiger
```

## 三日目 \* データベースの設計

---

### 機能が先かデータが先か

❗ポイント❗

- POA : 機能に商店を当てた設計法
- DOA : データに商店を当てた設計手法
- DOA : データベースの設計で用いる

### ■ POA

POA(Process Oriented Approach)

機能を中心に据えてシステム化したい業務を分析します。

データベースが誕生するまではこのPOAの設計手法が用いられ、

プログラムごとにファイルが作成されていた。

### ■ DOA

DOA(Data Oriented Approach)

データ中心に据えて業務を分析

データは複数のプログラムから共有されるもの考えます。

### ○ ERD

DOAではERD(Entity Relationship Diagram)と呼ばれる図を用いて分析や設計の結果を表します。

## 参考：OOA

オブジェクトに着目したOOA(Object Oriented Approach)という分析／設計手法もある

## データベースができるまで

### POINT

- 概念設計
- 論理設計
- 物理設計
- 運用設計

## 三日目 \* 2.データモデリング

---

### 2-1.ER図の読み方

#### ■表記法

- EDEF1X(ICAM DEFinition 1 eXtended)  
IDEF1X(アイデフワンエックス)
- IE(Infomation Engineering)表記法

#### ■エンティティ

エンティティ(Entity)は「実体」という意味  
図に表記するのは実際のデータではなくデータの入れ物  
実際のデータの例をインスタンスと言いますが、図には表記しません。  
エンティティが表  
インスタンスが行

#### ■カーディナリティ

親エンティティの1つのインスタンスに関連付けられる子エンティティのインスタンス数を表し、子エンティティ側に表記します

- 0以上
- 1以上(子側に「P」を表記)
- 0または1(子側に「Z」を表記)
- 定数(子側に定数を表記)

#### ■オプション

子エンティティの1つのインスタンスに関連付けられる親エンティティが0(ゼロ)でもよいことを表し、親エンティティ側に菱形を表記します。

## データベース・サーバー

---

### 1-1.ハードディスクには何があるの？

#### ■データファイル

- \*ブロック : 一番小さい区画
- \*エクステント: ブロックの集まり
- \*セグメント : エクステントの集まり

##### ●ブロック

表に挿入された行はブロックに格納されます。

行を検索したりすると対象となる行がブロック単位で読み込まれ、変更された行はブロック単位でデータファイルに書き込まれます。

##### ●エクステント

1つ以上のブロックの集まりがエクステント。

データベースに表を作成すると、データファイル内で開いているエクステントが割り当てられます。表にデータが追加され領域が不足すると、追加のエクステントが割り当てられます。

##### ●セグメント

表に割り当てられたエクステントを纏めたもの。

表が使用する領域を管理するための単位。

#### ■表領域

- \*USERS表領域  
ユーザーが作成する表を格納する為の領域
- \*SYSTEM表領域  
RDBMSが内部的に使用する表を格納するための領域

#### ■REDOログファイル

##### \*動作

REDOログファイルは複数個用意されている

1つ目がいっぱいになると2つ目が使用され、いっぱいになると3つ目が3つしかない場合、3つ目がいっぱいになると1つ目に上書きされる

### 1-2.メモリには何がある？

□ORACLEデータベース

- データベース
  - データファイル
  - REDOログファイル
- インスタンス
  - バックグラウンドプロセス
  - SGA

## SGA(System Global Area)

### ○データベースバッファキャッシュ

データファイルから読み込まれたデータが置かれる場所です。

ブロックをおいておけば、同じブロック内のデータが必要になった時に、データファイルにアクセスする必要が無いため処理時間を短縮可能

### ○REDOログバッファ

データベースバッファキャッシュに置かれたデータを変更する前に、REDOレコードが生成され、REDOログバッファに置かれます。

確定時にREDOログファイルに書き込まれ、保存されます。

## バックグラウンドプロセス

メモリ上にロードされた実行中のプログラム

データベースとメモリ間でデータの読み書きを行うために、複数のバックグラウンドプロセスが動作します。

用語：プロセス

メモリー内に読み込まれた実行状態のプログラムです。

バックグラウンドプロセスはユーザーによる操作は不要で、自動的に様々な処理を実行するプロセス

### ○ログライター

REDOログバッファに置かれたREDOレコードをREDOログファイルへ書き込む処理を担当します。

### ○データベースライター

データベースバッファキャッシュに置かれた変更済みのブロックをデータファイルに書き込む処理を担当します。

## 2.データ型

### 2-1.データの入れ物

#### ■データ型

\*数値

• NUMBER型

数値3桁であれば

NUMBER(3,0)

\*文字列

- ・ CHAR型

固定長文字列に適している

4文字なら

CHAR(4)

- ・ VARCHAR2型

サイズが一定でない文字列に適している

例えばサイズが4文字の文字列であれば

VARCHAR2(4)と定義 全角なら2文字

4バイトに満たない文字列を指定した場合、

必要な分しか領域を使用しない為無駄がありません。

- \* 日付／時刻型

- ・ DATE型

- ・ TIMESTAMP型

構文：DATA型

DATE

構文：TIMESTAMP型

TIMESTAMP(秒以下の桁数)

■NULLの扱い

必ず値がないといけない項目は

NOT NULL指定をするよと。

## 5日目 \* データベースの基本操作(1)

---

### SQLってなんだろう？

#### ■SQLの概要

1970年：IBMによって開発されたデータベース操作言語(SEQUEL：シークル)をベース

1986年：ANSIにより標準化

1987年：ISOにより国際標準化として規格が策定

↓

2008年：SQL:2008が最新

市場に出回るRDBMS製品の多くはANSI/ISO標準SQLに準拠していますが、それぞれ独自の機能を追加指定します。

Oracle Databaseは PL/SQL と呼ばれる言語を使用し、SQLの機能を拡張しています。

#### ■SQL文

○データ定義言語(DDL)

CREATE TABLE

DROP TABLE

○データ操作言語(DML)

SELECT

INSERT

UPDATE

DELETE

○データ制御言語(DCL)

GRANT

REVOKE

## SQL文の記述

○予約語／表名／列名は大文字小文字を区別しない

○予約語や記号は半角で記述する

○インデントや改行を入れることができる

○文字列、日付／時刻は「'」で囲む

○コメントは「--」の右側か「/\*」 「\*/」内に記述

⚠注意⚠

SQL\*Plusではセミコロンを末尾につけることで、  
一つのSQL文とみなす

## 1-2.表を作成してみよう

### コラム

DESCRIBEコマンド：表定義の確認

DESCと省略することも可能

## 1-3.データを検索してみよう

■SELECT文

## 5日目 \* データベースの基本操作(2)

---

■INSERT

■UPDATE

## 6日目 \* データベースの応用操作(1)

---

略

## 6日目 \* データベースの応用操作(2)

---

略



7日目\*略

おそらくSQLの試験で勉強する範囲とかぶる為。。。

**以上!!**