

IT 資格 道 場

ORACLE MASTER Silver SQL Oracle AI Database 26ai 学習用テキスト

実行して結果を覚える —— 読んで理解する1冊目

無料サンプル(第1章まで)

Oracle Japan「Oracle AI Database 26ai SQL Associate 1Z0-171-JPN」試験内容チ
ェックリスト(11分野) 準拠
2026年10月版

ご利用にあたって

- このテキストは「IT資格道場」(www.shikaku-doujou.com)の購入者限定テキストのうち、学習用テキストの第1章までを収録した**無料サンプル**です。どなたでもご覧いただけます。
- 無料サンプルも著作物です。転載・改変しての再配布・二次販売はできません。
- AI モデルの学習データとしてのご利用(学習目的の入力・データセット化を含む)はできません。
- 製品版(学習用テキスト・頻出度別ノート・暗記用ノートの3点)は、ご購入者を識別できる透かし入りのPDFとして提供されます。
- 本PDFはスマートフォン・タブレットでの閲覧に合わせたA5版面・文字サイズで組版しています。

目次

この教材群の使い方(3点セット)

この試験が問うもの

第1章 リレーショナル・データベースの概念 [1.1/1.2]

表・行・列・主キー・外部キーの用語 [1.1]

論理構造と物理構造 [1.1]

SQL 文の種類 (DDL・DML・DCL・TCL) [1.1]

リレーションシップ (1対多・多対多) と結合列 [1.1]

ER 図の要素と表の対応 [1.2]

ER 図の線を結合条件にする [1.2]

サンプルはここまでです

この教材群の使い方(3点セット)

種類	役割
① 学習用テキスト	これだけで問題が解けるようになる本編
② 頻出度別ノート	テキストを読んだら次はこれ。頻出順に絞った問題と解説で「解ける」に橋渡し。試験直前の最終チェックにも
③ 暗記用ノート	覚えるべき要点だけを一問一答に凝縮。空き時間の反復と用語の再確認用

使い方: ① 学習用を読む → ② 頻出度別で解き方を掴む → ③ 問題演習で腕試し → ④ 頻出度別に戻って再確認(試験直前もここ)。暗記用は空き時間や用語の再確認に。

この試験が問うもの

ORACLE MASTER Silver SQL Oracle AI Database 26ai (試験番号 1Z0-171-JPN) は、Oracle Database の SQL を、結果を読み取れる水準で問う試験です。[^1]SQL 文を見せて「実行するとどうなるか」「どの文がエラーになるか」を答えさせる問題が大半を占めます。

項目	内容
運営	Oracle Japan (試験の配信は Pearson VUE)
出題数	60問
試験時間	120分
合格ライン	65% (60問のうち39問の正解が目安)
出題形式	選択問題 (択一と、2つ選ぶ複数選択)

分野別の出題数は公式に公表されていません。公式の試験内容チェックリストは11分野に分かれていて、このテキストも同じ順番で11章に並べました。各章の例は、Oracle AI Database 26ai Free (バージョン 23.26.3) で実際に実行した結果です。SQL の下にある出力は、そのときの画面そのままです。

この教材で使う表は、第2章の冒頭で一度だけ作ります。社員表 EMPLOYEES (8行) と部署表 DEPARTMENTS (4行) です。後の章の例も、特に断らない限り同じ表を使い、作り直しません。日付の表示形式は、例を見やすくするために YYYY-MM-DD に揃え、NULL は (null) と表示する設定 (SET NULL) にしています。

第1章 リレーショナル・データベースの概念 [1.1/1.2]

表・行・列・主キー・外部キーの用語 [1.1]

リレーショナル・データベースは、データを表（テーブル）に入れて管理します。表は行と列でできています。行は1件分のデータです（レコードとも呼びます）。列は項目で、属性とも呼びます。社員表なら、1人の社員が1行で、姓や給与がそれぞれ列です。

行を1つに決める列が主キー（PRIMARY KEY）です。主キー制約は「NOT NULL かつ一意」を1つにまとめた制約で、1つの表に1つしか付けられません。社員番号と部署番号を組み合わせで1つの主キーにする複合主キーも作れます。

似た名前の一意キー（UNIQUE）は、重複を禁止しますが NULL は入れられます。主キーとの違いは NULL を許すかどうかで、一意キーは1つの表に何個でも付けられます。外部キー（FOREIGN KEY）は、別の表の主キーや一意キーを参照する列です（参照先に無い値は入れられません）。社員表の DEPARTMENT_ID は、部署表の主キーを参照する外部キーです。外部キーの列は NULL を許せば、参照先を持たない行も登録できます。チェック制約（CHECK）は、列に入れられる値を式で限定します。

制約の種類は、データ・ディクショナリの USER_CONSTRAINTS の CONSTRAINT_TYPE 列で分かります（主キーは P、外部キーは R、チェック制約と NOT NULL は C、一意キーは U）。

```
SELECT constraint_name, constraint_type, table_name
FROM user_constraints
WHERE constraint_type IN ('P', 'R')
ORDER BY constraint_name;
```

```
CONSTRAINT_NAME  C TABLE_NAME
```

```
-----
```

```
DEPT_PK          P DEPARTMENTS
```

```
EMP_DEPT_FK      R EMPLOYEES
```

```
EMP_MGR_FK       R EMPLOYEES
```

```
EMP_PK           P EMPLOYEES
```

```
4 rows selected.
```

論理構造と物理構造 [1.1]

SQL が扱うのは論理構造です。表、列、索引、ビューなどが論理構造に当たり、表がどのファイルのどの位置に置かれるかは、利用者が指定しません。Oracle が管理します。物理構造は OS 上のファイルで、データファイル、制御ファイル、REDO ログファイルがあります。

論理構造と物理構造をつなぐのが表領域 (テーブルスペース) です。表は表領域の中に作られ、表領域は1つ以上のデータファイルでできています。表を作るたびに専用のデータファイルができるわけではなく、表1つがファイル1つに対応するわけでもありません。1つのデータファイルには複数の表のデータが入ります。SQL でデータファイルを直接読み書きすることはできません。行を読むときは、必ず表を経由します。

```
SELECT tablespace_name, contents FROM user_tablespaces ORDER BY  
tablespace_name;
```

```
TABLESPACE_NAME  CONTENTS  
-----  
USERS            PERMANENT  
1 row selected.
```

SQL 文の種類(DDL・DML・DCL・TCL) [1.1]

SQL 文は働きによって4つに分けます。この分類は試験で頻繁に使われます。

分類	主な文	働き
DDL(データ定義言語)	CREATE・ALTER・DROP・ TRUNCATE・RENAME・ COMMENT	オブジェクトの作成・変更・ 削除
DML(データ操作言語)	SELECT・INSERT・ UPDATE・DELETE・MERGE	データの参照と変更
DCL(データ制御言語)	GRANT・REVOKE	権限の付与と取り消し
TCL(トランザクション制御 言語)	COMMIT・ROLLBACK・ SAVEPOINT	変更の確定と取り消し

見落としやすいのは TRUNCATE TABLE です(行を消す文なので DML に見えますが、DDL に属します)。DDL は実行した時点で暗黙的にコミットされるため、ROLLBACK では戻せません。DELETE は DML なので、COMMIT の前なら ROLLBACK で戻せます。DCL の GRANT と REVOKE も、DDL と同様に暗黙的にコミットされます。

リレーションシップ(1対多・多対多)と結合列 [1.1]

表と表のつながりをリレーションシップと呼びます。1つの部署に複数の社員が所属する関係は、1対多です。表に表すときは、「多」の側の表に外部キーを置き、「1」の側には置きません。社員表に DEPARTMENT_ID 列を持たせ、部署表の主キーを参照させます。部署表の側に社員番号を並べる列を作る方法や、1つの列に複数の値をまとめて入れる方法は、リレーショナルの形になりません。

学生と科目のように、どちら側も複数と結び付く関係は多対多です。1つの外部キーでは表せないで、交差表(結合表)を作ります。交差表は学生番号と科目番号を持ち、2つの列の組合せを複合主キーにして、それぞれを外部キーにします。多対多を、2つの1対多に分解する形です。成績や履修日など、関係そのものが持つ情報は、交差表にしか置けません。

同じ表の中で結び付く関係もあります。社員表の MANAGER_ID には、同じ表の EMPLOYEE_ID の値が入ります。これは自己参照の外部キーで、再帰的な関係とも呼びます。MANAGER_ID は複数の部下で同じ値が並ぶので、主キーにはなりません。社長のように上司がいない行は MANAGER_ID が NULL で、外部キーの違反にはなりません。

親の行を削除するときの動きは、外部キーの指定で決まります。ON DELETE の指定が無いと、子の行が残っている親は削除できず、ORA-02292 のエラーになります。子の行の削除は、いつでも可能です。

```
DELETE FROM departments WHERE department_id = 20;
```

```
DELETE FROM departments WHERE department_id = 20
```

*

ERROR at line 1:

ORA-02292: integrity constraint (STUDY.EMP_DEPT_FK) violated - child record found

ON DELETE CASCADE を付けると子の行も一緒に消え、ON DELETE SET NULL を付けると子の外部キー列が NULL になります (詳しくは第9章の制約の節)。

ER 図の要素と表の対応 [1.2]

ER 図 (エンティティ・リレーションシップ図) は、データベースの論理設計を描く図で、行のデータは描きません。図の要素と、データベースの要素は次のように対応します。

ER 図	データベース
エンティティ	表
属性	列
インスタンス	行
UID (一意識別子)	主キー
リレーションシップ	外部キー、結合条件

UID は、インスタンスを1つに決める属性です。主キーに対応します。UID の候補が複数ある場合は、主キーに選ばなかった候補を一意キー制約として残せます。リレーションシップは外部キーに対応し、索引やビューには対応しません。

ER 図の線を結合条件にする [1.2]

ER 図の線は、SELECT 文の結合条件に対応します。部署から社員へ「1対多」の線が引かれ、両端に DEPARTMENT_ID が示されているなら、ON 句の条件は両端の列を等しいとする式です。

```
DEPARTMENTS 1 -----< EMPLOYEES
(department_id)      (department_id)
```

```
SELECT d.department_name, e.last_name
FROM departments d JOIN employees e
    ON d.department_id = e.department_id
ORDER BY e.employee_id;
```

DEPARTMENT_NAME	LAST_NAME
Admin	Sato
Sales	Suzuki
Support	Tanaka
Sales	Ito
Sales	Watanabe
Support	Yamada
Support	Kato

7 rows selected.

線の両端にない列で結合しても、関係は表せません。部署名と姓を等しいとする条件や、部署番号と社員番号を等しいとする条件は、意味の違う列を比べ

ているだけです。CROSS JOIN は条件を付けずに全組合せを作るので、線の関係を表しません。外部キー制約があっても、結合条件は自動では補われません。ON 句か USING 句で書く必要があります。

自己参照の線は、同じ表に別名を2つ付けて結合します。部下の側の MANAGER_ID と、上司の側の EMPLOYEE_ID を等しいとします。同じ表を2回使うときは別名が必須で、別名が無いと列の参照があいまいになってエラーになります。社員番号どうしを等しいとすると、各社員が自分自身と組になるだけで、上司は出ません。

```
SELECT w.last_name AS worker, m.last_name AS manager
FROM employees w JOIN employees m
    ON w.manager_id = m.employee_id
ORDER BY w.employee_id;
```

WORKER	MANAGER
Suzuki	Sato
Tanaka	Sato
Ito	Suzuki
Watanabe	Suzuki
Yamada	Tanaka
Kato	Tanaka
Hayashi	Sato

7 rows selected.

社長の Sato は MANAGER_ID が NULL なので、内部結合の結果に現れません。含めたい場合は外部結合にします。

ER 図に出てくる表がすべて線で結ばれているとは限りません。給与の下限と上限を持つ等級表のように、他の表と線で結ばれていない参照用の表もあります。この表からは、給与が下限と上限の間にあるという条件で等級を引きます。線のある表は外部キーと主キーの等結合、線の無い範囲の表は非等価結合です。

```
CREATE TABLE job_grades (grade_level CHAR(1), lowest_sal NUMBER,  
highest_sal NUMBER);  
INSERT INTO job_grades VALUES ('A', 0, 2999);  
INSERT INTO job_grades VALUES ('B', 3000, 5999);  
INSERT INTO job_grades VALUES ('C', 6000, 99999);  
SELECT e.last_name, d.department_name, g.grade_level  
FROM employees e  
      JOIN departments d ON e.department_id = d.department_id  
      JOIN job_grades g ON e.salary BETWEEN g.lowest_sal AND g.highest_sal  
ORDER BY e.employee_id;
```

Table created.

1 row created.

1 row created.

1 row created.

LAST_NAME	DEPARTMENT_NAME	G
-----------	-----------------	---

-----	-----	-
-------	-------	---

Sato	Admin	C
------	-------	---

Suzuki	Sales	C
--------	-------	---

Tanaka	Support	B
--------	---------	---

Ito	Sales	B
-----	-------	---

Watanabe	Sales	A
----------	-------	---

Yamada	Support	A
--------	---------	---

Kato	Support	A
------	---------	---

7 rows selected.

内部結合は、両方の表に対応する行がある組合せだけを返し、対応する行が無い部署は返しません。社員のいない部署も一覧に残したい場合は、残したい部署表を左に置いて LEFT OUTER JOIN にします。社員がいない部署は、姓が NULL で出ます。CROSS JOIN に WHERE を付ける形は内部結合と同じ結果で、社員のいない部署は残りません。RIGHT OUTER JOIN は右側の社員表の全行を残すので、部署に所属しない社員が残り、社員のいない部署は消えます。

```
SELECT d.department_name, e.last_name
FROM departments d LEFT OUTER JOIN employees e
  ON d.department_id = e.department_id
ORDER BY d.department_id, e.employee_id;
```

DEPARTMENT_NAME	LAST_NAME
-----------------	-----------

Admin	Sato
Sales	Suzuki
Sales	Ito
Sales	Watanabe
Support	Tanaka
Support	Yamada
Support	Kato
Planning	(null)

8 rows selected.

サンプルはここまでです

続き(第2章～巻末)は、ORACLE MASTER Silver SQL Oracle AI Database 26ai 問題集のご購入で、頻出度別ノート・暗記用ノートと合わせて3点すべてダウンロードできます。

SAMPLE

ORACLE MASTER Silver SQL Oracle AI Database 26ai 学習用テキスト(無料サンプル)

版

2026年10月版(Oracle Japan「Oracle AI Database 26ai SQL Associate 1Z0-171-JPN」試験内容チェックリスト(11分野) 準拠)

発行

IT資格道場(<https://www.shikaku-doujou.com>)

お問い合わせ

info@shikaku-doujou.com

Oracle、Java、MySQL および NetSuite は、Oracle、その子会社および関連会社の米国およびその他の国における登録商標です。ORACLE MASTER は、Oracle が日本で提供する認定資格の名称です。記載されているその他の会社名・製品名・サービス名は、各社の商標または登録商標です。

本教材は公式の試験内容チェックリストと Oracle AI Database 26ai Free で実行した結果を基に作成したもので、Oracle およびその子会社・関連会社との提携・承認関係はありません。

本文・図表・設問はすべてオリジナルの著作物です。無断での転載・複製・再配布・二次販売はできません。お気づきの点は info@shikaku-doujou.com までご連絡ください。