

ORACLE®

**ORACLE MASTER Bronze**  
**Oracle Database 11g**  
**スキルチェック解説**

日本オラクル株式会社  
オラクルユニバーシティ



以下の事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメント(確約)するものではないため、購買決定を行う際の判断材料になさらないで下さい。オラクル製品に関して記載されている機能の開発、リリースおよび時期については、弊社の裁量により決定されます。

OracleとJavaは、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。

# スキルチェック解説

SQL編解説

Bronze DBA編解説

# SQL解説

# EMPLOYEES表

EMPNO	LAST_NAME	FIRST_NAME	SALARY	DEPTNO
1	AOYAMA	TAKASHI	20	20
2	AKITA	TSUYOSHI	30	10
3	AKAGAWA	ICHIRO	30	40
4	YAMAUCHI	JIRO	20	40
5	OYAMADA	SHO	20	20
6	KITAHARA	GORO	.	10
7	<u>koyama</u>	ken	.	10

# 1) 苗字と名前を連結した文字列をFull Nameという見出しで、給与を12倍した金額をYearlyという見出しで表示したい

- A) `SELECT last_name+first_name as "Full Name", salary*12 as "Yearly" FROM employees;`
- B) `SELECT last_name||first_name as "Full Name", salaryx12 "Yearly" FROM employees;`
- C) `SELECT last_name, first_name as "Full Name", salary*12 as "Yearly" FROM employees;`
- D) `SELECT last_name||first_name "Full Name", salary*12 "Yearly" FROM employees;`
- E) 正しいものはない

# 1) 苗字と名前を連結した文字列をFull Nameという見出しで、給与を12倍した金額をYearlyという見出しで表示したい

- A) `SELECT last_name+first_name as "Full Name", salary*12 as "Yearly" FROM employees;`
- B) `SELECT last_name||first_name as "Full Name", salaryx12 "Yearly" FROM employees;`
- C) `SELECT last_name, first_name as "Full Name", salary*12 as "Yearly" FROM employees;`
- D) `SELECT last_name||first_name "Full Name", salary*12 "Yearly" FROM employees;`
- E) 正しいものはない

## 2) 部門番号と苗字を、部門番号の大きい順、同じ部門番号の中では苗字のABC順で表示したい

- A) `SELECT deptno, last_name FROM employees ORDER BY deptno DESC, last_name;`
- B) `SELECT deptno, last_name FROM employees ORDER BY deptno, last_name ASC;`
- C) `SELECT deptno, last_name FROM employees ORDER BY deptno ASC, last_name DESC;`
- D) `SELECT deptno, last_name FROM employees ORDER BY deptno AND last_name;`
- E) 正しいものはない



## 2) 部門番号と苗字を、部門番号の大きい順、同じ部門番号の中では苗字のABC順で表示したい

- A) `SELECT deptno, last_name FROM employees ORDER BY deptno DESC, last_name;`
- B) `SELECT deptno, last_name FROM employees ORDER BY deptno, last_name ASC;`
- C) `SELECT deptno, last_name FROM employees ORDER BY deptno ASC, last_name DESC;`
- D) `SELECT deptno, last_name FROM employees ORDER BY deptno AND last_name;`
- E) 正しいものはない

### 3) システム日付を「木曜日」のようなフォーマットで表示したい

- A) `SELECT TO_CHAR(sysdate, 'D') FROM dual;`
- B) `SELECT TO_CHAR(sysdate, 'Day') FROM dual;`
- C) `SELECT TO_CHAR(timestamp, 'Day') FROM system;`
- D) `SELECT TO_CHAR(systime, 'Day') FROM system;`

### 3) システム日付を「木曜日」のようなフォーマットで表示したい

- A) `SELECT TO_CHAR(sysdate, 'D') FROM dual;`
- B) `SELECT TO_CHAR(sysdate, 'Day') FROM dual;`
- C) `SELECT TO_CHAR(timestamp, 'Day') FROM system;`
- D) `SELECT TO_CHAR(systime, 'Day') FROM system;`

4) salaryの値がNULLとなっている社員が存在する  
**SELECT last\_name, salary FROM employees  
WHERE salary != NULL ORDER BY salary DESC;**

- A) salaryがNULLの社員は検索結果の一番上に表示される
- B) salaryがNULLの社員は検索結果の一番下に表示される
- C) salaryがNULLの社員は検索結果に含まれない
- D) 検索結果は一行も表示されない
- E) SQLの構文エラーが表示される

4) salaryの値がNULLとなっている社員が存在する  
**SELECT last\_name, salary FROM employees  
WHERE salary != NULL ORDER BY salary DESC;**

- A) salaryがNULLの社員は検索結果の一番上に表示される
- B) salaryがNULLの社員は検索結果の一番下に表示される
- C) salaryがNULLの社員は検索結果に含まれない
- D) 検索結果は一行も表示されない
- E) SQLの構文エラーが表示される

## 5) 苗字がAで始まり、かつ、部門20もしくは部門40に所属している全社員を検索したい

- A) WHERE last\_name LIKE 'A%' AND deptno=20 OR deptno=40;
- B) WHERE last\_name LIKE 'A\_' AND deptno LIKE '\_0';
- C) WHERE last\_name LIKE 'A\*' AND deptno in (20, 40);
- D) WHERE last\_name = 'A%' AND ( deptno=20 OR deptno=40);
- E) 正しいものはない

## 5) 苗字がAで始まり、かつ、部門20もしくは部門40に所属している全社員を検索したい

- A) WHERE last\_name LIKE 'A%' AND deptno=20 OR deptno=40;
- B) WHERE last\_name LIKE 'A\_' AND deptno LIKE '\_0';
- C) WHERE last\_name LIKE 'A\*' AND deptno in (20, 40);
- D) WHERE last\_name = 'A%' AND ( deptno=20 OR deptno=40);
- E) 正しいものはない

## 6) 大文字、小文字に関係なく、'yama'という文字列が苗字に入っている社員を検索したい

- A) `WHERE initcap(last_name) LIKE '%Yama%';`
- B) `WHERE last_name LIKE '%yama' OR last_name like '%YAMA';`
- C) `WHERE upper(last_name) LIKE '%YAMA%';`
- D) `WHERE last_name LIKE '%yama%' AND last_name like '%YAMA%';`
- E) 正しいものはない



## 6) 大文字、小文字に関係なく、'yama'という文字列が苗字に入っている社員を検索したい

- A) WHERE initcap(last\_name) LIKE '%Yama%';
- B) WHERE last\_name LIKE '%yama' OR last\_name like '%YAMA';
- C) WHERE upper(last\_name) LIKE '%YAMA%';
- D) WHERE last\_name LIKE '%yama%' AND last\_name like '%YAMA%';
- E) 正しいものはない

## 7) CURR\_ORDER表のCUSTOMER\_ID列と CUSTOMER表のID列で結合をしようと思いますが、 CUSTOMER表のID列に含まれるNULL値も表示

- A) 外部結合
- B) 自己結合
- C) 自然結合
- D) 等価結合

## 7) CURR\_ORDER表のCUSTOMER\_ID列と CUSTOMER表のID列で結合をしようと思いますが、 CUSTOMER表のID列に含まれるNULL値も表示

- A) 外部結合
- B) 自己結合
- C) 自然結合
- D) 等価結合

## 8) 部門の平均給与が会社全体の平均給与より高い部門を検索したい

- A) `SELECT deptno, AVG(salary) FROM employees WHERE AVG(salary) > (SELECT AVG(salary) FROM employees);`
- B) `SELECT deptno, AVG(salary) FROM employees WHERE AVG(salary) > (SELECT AVG(salary) FROM employees) GROUP BY deptno;`
- C) `SELECT deptno, AVG(salary) FROM employees GROUP BY deptno HAVING AVG(salary) > (SELECT AVG(salary) FROM employees);`
- D) `SELECT deptno, AVG(salary) FROM employees WHERE AVG(salary) > AVG(employees.salary);`
- E) 正しいものはない

## 8) 部門の平均給与が会社全体の平均給与より高い部門を検索したい

- A) `SELECT deptno, AVG(salary) FROM employees WHERE AVG(salary) > (SELECT AVG(salary) FROM employees);`
- B) `SELECT deptno, AVG(salary) FROM employees WHERE AVG(salary) > (SELECT AVG(salary) FROM employees) GROUP BY deptno;`
- C) `SELECT deptno, AVG(salary) FROM employees GROUP BY deptno HAVING AVG(salary) > (SELECT AVG(salary) FROM employees);`
- D) `SELECT deptno, AVG(salary) FROM employees WHERE AVG(salary) > AVG(employees.salary);`
- E) 正しいものはない

**9) SELECT first\_name, last\_name FROM physician  
WHERE physician\_id NOT IN  
(SELECT physician\_id FROM physician WHERE license\_no =17852);**

- A) WHERE physician\_id <> ALL (SELECT physician\_id FROM physician WHERE license\_no = 17852);
- B) WHERE physician\_id <> ANY (SELECT physician\_id FROM physician WHERE license\_no = 17852);
- C) WHERE physician\_id IN (SELECT physician\_id FROM physician WHERE license\_no = 17852);
- D) WHERE license\_no = 17852;

**9) SELECT first\_name, last\_name FROM physician  
WHERE physician\_id NOT IN  
(SELECT physician\_id FROM physician WHERE license\_no =17852);**

- A) WHERE physician\_id <> ALL (SELECT physician\_id FROM physician WHERE license\_no = 17852);
- B) WHERE physician\_id <> ANY (SELECT physician\_id FROM physician WHERE license\_no = 17852);
- C) WHERE physician\_id IN (SELECT physician\_id FROM physician WHERE license\_no = 17852);
- D) WHERE license\_no = 17852;

10) projno number(3) primary key, project varchar2(20)  
not null, start\_date date, leader varchar2(20)

この表に、projno 30, project システム移行, start\_date 挿入  
する日の日付(時刻は切捨て), leader NULL を挿入したい

- A) INSERT INTO PROJECT (projno, project, start\_date, leader)  
VALUES(30, 'システム移行', trunc(sysdate, 'dd'), ' ');
- B) INSERT INTO PROJECT (projno, project, start\_date, leader)  
VALUES(30, 'システム移行', trunc(sysdate, 'dd'), 'NULL');
- C) INSERT INTO PROJECT (projno, project, start\_date, leader)  
VALUES('30', "システム移行", trunc(sysdate, 'dd'), NULL);
- D) INSERT INTO PROJECT VALUES(30, 'システム移行', trunc(sysdate,  
'dd'), NULL);
- E) 正しいものはない



# 11) DMLについて

- A) INSERT文は必ずVALUES句を含む必要がある
- B) UPDATE文は1つの表の複数の列を更新できる
- C) DELETE文は表の行や列を削除するために使用できる
- D) INTO句が指定されている場合、UPDATE文は表に行を追加する

# 11) DMLについて

- A) INSERT文は必ずVALUES句を含む必要がある
- B) UPDATE文は1つの表の複数の列を更新できる
- C) DELETE文は表の行や列を削除するために使用できる
- D) INTO句が指定されている場合、UPDATE文は表に行を追加する

## 12) TRUNCATE文で表のデータを切り捨てた

- A) TRUNCATE文もCOMMIT前であればロールバックできる
- B) TRUNCATE文でデータを切り捨てると、表そのものも完全に削除される
- C) TRUNCATE文での挙動はDELETE文と全く同じである
- D) TRUNCATE文はDDLなのでロールバックすることができない

## 13) シノニムについて

- A) 表に対してシノニムを作ると、元の表名をSQLで使用できなくなる
- B) シノニムを作ると権限が無くても他人のオブジェクトにアクセスできる
- C) シノニムは表の別名に過ぎず、シノニムの有無によって挙動が変わることはない
- D) シノニムを削除すると、対応するオブジェクトも同時に削除される

## 13) シノニムについて

- A) 表に対してシノニムを作ると、元の表名をSQLで使用できなくなる
- B) シノニムを作ると権限が無くても他人のオブジェクトにアクセスできる
- C) シノニムは表の別名に過ぎず、シノニムの有無によって挙動が変わることはない
- D) シノニムを削除すると、対応するオブジェクトも同時に削除される

# 14) SELECT line\_ seq.CURRVAL FROM dual;

- A) LINE\_SEQ順序の現在の値を表示する
- B) LINE\_SEQ順序に次の値を移入する
- C) LINE\_SEQ順序の次の値を表示する
- D) このような使用法はできない

# 14) SELECT line\_ seq.CURRVAL FROM dual;

- A) LINE\_SEQ順序の現在の値を表示する
- B) LINE\_SEQ順序に次の値を移入する
- C) LINE\_SEQ順序の次の値を表示する
- D) このような使用法はできない

## 15)ビューの作成に関して

- A) OR REPLACEオプション句を使用すると、同じ名前のビューがすでに存在していない場合エラーになる
- B) WITH CHECK OPTION句を使用する場合は制約の名前を指定する必要があり、指定しないと文が失敗する
- C) 列の別名を使用することで、ビューの列を実表の列とは別の名前にすることができる
- D) 実際の表が存在しない場合、ビューを作成することはできない



# 15)ビューの作成に関して

- A) OR REPLACEオプション句を使用すると、同じ名前のビューがすでに存在していない場合エラーになる
- B) WITH CHECK OPTION句を使用する場合は制約の名前を指定する必要があり、指定しないと文が失敗する
- C) 列の別名を使用することで、ビューの列を実表の列とは別の名前にすることができる
- D) 実際の表が存在しない場合、ビューを作成することはできない

# Bronze DBA解説

# 1) Oracleソフトウェアを配置するディレクトリを指定する環境変数として正しいものを選択して下さい

- A) CONTROL\_FILES
- B) DB\_FILE\_CREATE\_DEST
- C) ORACLE\_HOME
- D) ORACLE\_SID

# 1) Oracleソフトウェアを配置するディレクトリを指定する環境変数として正しいものを選択して下さい

- A) CONTROL\_FILES
- B) DB\_FILE\_CREATE\_DEST
- C) ORACLE\_HOME
- D) ORACLE\_SID

## 2) Enterprise Manager Database Controlのポート番号を忘れてしまった。何を見て確認すればよいか。

- A) 初期化パラメータ・ファイル
- B) V\$PORTLIST
- C) listener.oraファイル
- D) portlist.iniファイル

## 2) Enterprise Manager Database Controlのポート番号を忘れてしまった。何を見て確認すればよいか。

- A) 初期化パラメータ・ファイル
- B) V\$PORTLIST
- C) listener.oraファイル
- D) portlist.iniファイル

### 3) Enterprise Managerのブラックアウト期間はどういうケースで設定するか。

- A) チューニングのために通常より監視項目を増やしたいケースで使用する
- B) ディスプレイの故障時など、Enterprise Managerの画面は表示せず、Eメールを使用してのアラート通知のみを有効にしたいケースで使用する
- C) チューニングのために通常より監視頻度を上げたいケースで使用する
- D) ハードウェア・メンテナンスなどのために、監視を一時停止したいケースで使用する

### 3) Enterprise Managerのブラックアウト期間はどういうケースで設定するか。

- A) チューニングのために通常より監視項目を増やしたいケースで使用する
- B) ディスプレイの故障時など、Enterprise Managerの画面は表示せず、Eメールを使用してのアラート通知のみを有効にしたいケースで使用する
- C) チューニングのために通常より監視頻度を上げたいケースで使用する
- D) ハードウェア・メンテナンスなどのために、監視を一時停止したいケースで使用する



## 4) リスナーが停止するとどうなるか。

- A) クライアントからデータベースへの新規接続ができなくなる
- B) すでに接続されている全セッションが強制的に切断される
- C) 実行中のトランザクションがロールバックされる
- D) インスタンスが再起動される

## 4) リスナーが停止するとどうなるか。

- A) クライアントからデータベースへの新規接続ができなくなる
- B) すでに接続されている全セッションが強制的に切断される
- C) 実行中のトランザクションがロールバックされる
- D) インスタンスが再起動される

## 5) SMONの説明として正しいものを選択して下さい。

- A) クライアント (SQL Developerなど) が異常終了した際に、そのプロセスで使用していたリソースを解放し、プロセス・リカバリを行う
- B) インスタンス障害発生後の再起動の際に、インスタンス・リカバリを行う
- C) SGAメモリの使用状況を定期的に監視し、各コンポーネントへのメモリ割当てを動的に調整する
- D) メトリックを監視し、特定のメトリックがしきい値を超えるとアラートを発行する

## 5) SMONの説明として正しいものを選択して下さい。

- A) クライアント (SQL Developerなど) が異常終了した際に、そのプロセスで使用していたリソースを解放し、プロセス・リカバリを行う
- B) インスタンス障害発生後の再起動の際に、インスタンス・リカバリを行う
- C) SGAメモリの使用状況を定期的に監視し、各コンポーネントへのメモリ割当てを動的に調整する
- D) メトリックを監視し、特定のメトリックがしきい値を超えるとアラートを発行する

## 6) インスタンス起動のステップについて正しいものを選択して下さい。

- A) マウントするときにデータファイルがオープンされる
- B) マウントするときにチェックポイントが行われる
- C) マウントするときに制御ファイルが読み取られる
- D) マウントするときにインスタンス・リカバリが行われる

## 6) インスタンス起動のステップについて正しいものを選択して下さい。

- A) マウントするときにデータファイルがオープンされる
- B) マウントするときにチェックポイントが行われる
- C) マウントするときに制御ファイルが読み取られる
- D) マウントするときにインスタンス・リカバリが行われる

## 7) Enterprise Managerから停止モード「標準」でインスタンスを停止しようとしているが、いくら待っていても停止しない。

- A) そのまま待っていれば必ず停止するので、しばらく放置する
- B) 残っているセッションがないか確認して、そのセッションを強制的に切断する必要がある
- C) 未Commitのトランザクションがないか確認して、そのトランザクションをCommitする必要がある
- D) OSを強制終了する必要がある

## 7) Enterprise Managerから停止モード「標準」でインスタンスを停止しようとしているが、いくら待っていても停止しない。

- A) そのまま待っていれば必ず停止するので、しばらく放置する
- B) 残っているセッションがないか確認して、そのセッションを強制的に切断する必要がある
- C) 未Commitのトランザクションがないか確認して、そのトランザクションをCommitする必要がある
- D) OSを強制終了する必要がある



## 8) REDOログを多重化する際の注意点として正しいものを選択して下さい。

- A) ディスク障害に備え、同一グループを構成するメンバー同士は別々のドライブに格納する
- B) ディスク障害に備え、グループ毎に別々のドライブに格納する
- C) REDOログは多重化できない
- D) 初期化パラメータで複数のファイルを指定することにより多重化できる

## 8) REDOログを多重化する際の注意点として正しいものを選択して下さい。

- A) ディスク障害に備え、同一グループを構成するメンバー同士は別々のドライブに格納する
- B) ディスク障害に備え、グループ毎に別々のドライブに格納する
- C) REDOログは多重化できない
- D) 初期化パラメータで複数のファイルを指定することにより多重化できる

## 9) UNDO表領域について正しいものを選択して下さい。

- A) UNDO表領域はMOUNT状態でのみリカバリできる
- B) UNDO表領域はOPEN状態でもリカバリできる
- C) UNDO表領域は一時表領域同様、リカバリは不要である。
- D) UNDO表領域はどんな状態でもリカバリは出来ないので多重化しておく必要がある

## 9) UNDO表領域について正しいものを選択して下さい。

- A) UNDO表領域はMOUNT状態でのみリカバリできる
- B) UNDO表領域はOPEN状態でもリカバリできる
- C) UNDO表領域は一時表領域同様、リカバリは不要である。
- D) UNDO表領域はどんな状態でもリカバリは出来ないので多重化しておく必要がある

# 10) ユーザーを削除すると、そのユーザーが所有していたスキーマ・オブジェクト(表、索引など)はどうか。

- A) スキーマ・オブジェクトを所有しているユーザーは削除できない
- B) スキーマ・オブジェクトは保持され、SYSTEMユーザーの所有に変わる
- C) ユーザーを削除するときに、スキーマ・オブジェクトを保持するか指定できる
- D) そのユーザーが所有していたスキーマ・オブジェクトはデータベースから削除される

# 10) ユーザーを削除すると、そのユーザーが所有していたスキーマ・オブジェクト(表、索引など)はどうか。

- A) スキーマ・オブジェクトを所有しているユーザーは削除できない
- B) スキーマ・オブジェクトは保持され、SYSTEMユーザーの所有に変わる
- C) ユーザーを削除するときに、スキーマ・オブジェクトを保持するか指定できる
- D) そのユーザーが所有していたスキーマ・オブジェクトはデータベースから削除される

# 11) 10回ログインに失敗したら、デフォルトのパスワード・ポリシーによってユーザー・アカウントがロックされてしまった。

- A) 一旦ロックされると解除はできない
- B) DBAに連絡し、ロックを手動解除してもらう
- C) ログイン失敗によるユーザー・アカウント・ロックの仕組みはない
- D) 60分待てば自動でロック解除される

11) 10回ログインに失敗したら、デフォルトのパスワード・ポリシーによってユーザー・アカウントがロックされてしまった。

- A) 一旦ロックされると解除はできない
- B) DBAに連絡し、ロックを手動解除してもらう
- C) ログイン失敗によるユーザー・アカウント・ロックの仕組みはない
- D) 60分待てば自動でロック解除される



## 12) ロールについて正しいものを選択して下さい

- A) 一人のユーザーにつき、割当てできるロールは一つのみである
- B) システム権限もオブジェクト権限もロールに追加できる
- C) ロールに追加できるのはシステム権限と他のロールだけである
- D) ロールに別のロールを追加することはできない

## 12) ロールについて正しいものを選択して下さい

- A) 一人のユーザーにつき、割当てできるロールは一つのみである
- B) システム権限もオブジェクト権限もロールに追加できる
- C) ロールに追加できるのはシステム権限と他のロールだけである
- D) ロールに別のロールを追加することはできない

# 13) Enterprise Managerで推奨バックアップ計画を設定した。

- A) 毎晩フルバックアップを取得するような運用である
- B) 初めにバックアップセットを作成し、その後、毎晩、増分バックアップを新たに取得する運用である
- C) 初めにイメージコピーを作成し、その後、毎晩、前夜までの増分バックアップを適用しつつ、増分バックアップを新たに取得する運用である
- D) 週末ごとにフルバックアップを、平日深夜は増分バックアップを取得する運用である

# 13) Enterprise Managerで推奨バックアップ計画を設定した。

- A) 毎晩フルバックアップを取得するような運用である
- B) 初めにバックアップセットを作成し、その後、毎晩、増分バックアップを新たに取得する運用である
- C) 初めにイメージコピーを作成し、その後、毎晩、前夜までの増分バックアップを適用しつつ、増分バックアップを新たに取得する運用である
- D) 週末ごとにフルバックアップを、平日深夜は増分バックアップを取得する運用である

# 14) バックアップ・ポリシーを設定する際の、リカバリ・ウィンドウとは何か。

- A) リカバリ専用の時間帯のことである
- B) リカバリ設定をするための、Enterprise Managerの別ウィンドウのことである
- C) リカバリの所要時間のことである
- D) 不完全リカバリで何日前の状態まで戻せるか、その期間を指定する

# 14) バックアップ・ポリシーを設定する際の、リカバリ・ウィンドウとは何か。

- A) リカバリ専用の時間帯のことである
- B) リカバリ設定をするための、Enterprise Managerの別ウィンドウのことである
- C) リカバリの所要時間のことである
- D) 不完全リカバリで何日前の状態まで戻せるか、その期間を指定する

# 15) ADDMの動作について正しいものを選択して下さい。

- A) データベース時間を最も消費するコンポーネントおよび操作に集中して分析する
- B) トップダウンのシステム分析を10分に1回実行する
- C) SMONとPMONによって取得されたデータを元に分析する
- D) ADDMは実行されたSQLについてのアドバイザであり、システム全体についてのアドバイスは出さない

# 15) ADDMの動作について正しいものを選択して下さい。

- A) データベース時間を最も消費するコンポーネントおよび操作に集中して分析する
- B) トップダウンのシステム分析を10分に1回実行する
- C) SMONとPMONによって取得されたデータを元に分析する
- D) ADDMは実行されたSQLについてのアドバイザであり、システム全体についてのアドバイスは出さない



# 勉強法

# SQL

- 分からない箇所はそのままにせずに実機で試して理解を深めましょう
- 後半で表の作成方法を学べば、確認用のデータを自分で用意できます
- 勉強用にデータベースをインストールすることが難しい場合は以下を使うと便利です

<http://apex.oracle.com/pls/apex/>

# Bronze DBA

- 試験範囲全体を網羅しましょう
- 勉強用にデータベースをインストールし、実機で確認すると理解が深まります
- Bronze DBAに合格したら、その勢いでSilverにチャレンジされるとよいでしょう

**Hardware and Software**

**ORACLE®**

**Engineered to Work Together**

**ORACLE®**