

SQL トレーニング

～研修受講後のスキルアップサポート～

対応バージョン : Oracle 10gR1 ~ 12cR1

本資料は、アシスト Oracle 研修をご受講いただいたお客様からのご質問や、研修ではご案内できなかった情報などを FAQ にまとめたものです。研修受講後のスキルアップの一助として、是非お役立てください。

※ご利用上の注意事項は最後のページにまとめられております。ご確認のうえ、ご利用ください。

第 2 章 問合せの基本操作

1	<p>Q. 検索結果は何をもとに並べられているのですか。</p> <p>A. 行が物理的に格納されている順番で返されます。 行を DELETE した場合、その空いた領域に新規に INSERT した行が格納される場合があるので、必ずしも新しい行が物理的に最後に追加されるとは限りません。</p> <p>検索結果の順序を制御するには ORDER BY 句を使用してください。</p>
2	<p>Q. 一意な値が何件あるか確認したい場合はどのような SQL 文を実行すればよいですか。</p> <p>A. データの件数を求める COUNT 関数と、値を一意にして表示できる DISTINCT 句を組み合わせて使用します。</p> <p><例>会社に職種がいくつ存在しているか表示する。 SQL> SELECT COUNT (DISTINCT (JOB)) FROM EMP;</p> <pre> COUNT (DISTINCT (JOB)) ----- 5 </pre>
3	<p>Q. ESCAPE 文字として使用できない文字を教えてください。</p> <p>A. ESCAPE 文字は基本的に長さが 1 バイトであれば指定することができますが、中には使用できない文字も存在します。使用できない主な文字は以下のとおりです。</p> <ol style="list-style-type: none"> 1. 長さが 2 バイト以上の文字 (全角のひらがななど) 2. 文字パターン内で使用されている文字 3. &(' &' は SQL*Plus では置換関数を意味します)
4	<p>Q. SQL*Plus の DESCRIBE コマンド以外で、表の列構造を確認することはできますか。</p> <p>A. USER_TAB_COLUMNS ビューを使用すると、表の各列に関する詳細情報を確認できます。</p> <p>DESCRIBE の情報はビューの以下の列でご確認いただけます。</p> <ul style="list-style-type: none"> ・ COLUMN_NAME 列名 ・ DATA_TYPE データ型 ・ DATA_LENGTH データ長 ・ DATA_SCALE 数値の小数点以下の桁数 ・ NULLABLE NULL 定義の可否を表示

5	Q.	選択リストにアスタリスク(*)を使用して、位置に基づくソートを行うとどうなりますか。														
	A.	その表を構成している列の順序に対応して、暗黙的に番号が割り振られます。														
6	Q.	文字データが日本語の場合、ソートは五十音順に行われるのですか。														
	A.	<p>五十音順ではなく、文字コード順になります。 文字コードの種類はマシンによって異なりますが、主なものとして ASCII コードがあります。 Oracle では ASCII 関数を使用してコード数を調べることができます。</p> <p><例>NAME 列のデータの文字コードを確認する。 SQL> SELECT name, ASCII(name) FROM test 2 ORDER BY name ASC;</p> <table border="1"> <thead> <tr> <th>NAME</th> <th>ASCII (NAME)</th> </tr> </thead> <tbody> <tr> <td>SMITH</td> <td>83</td> </tr> <tr> <td>わたなべ XXX</td> <td>33517</td> </tr> <tr> <td>伊藤 XXX</td> <td>35017</td> </tr> </tbody> </table> <p>なお、五十音順のソートを行いたい場合は、初期化パラメータ NLS_SORT を設定する必要があります。 ※初期化パラメータ NLS_SORT の詳細については「リファレンス」マニュアルをご参照ください。</p>	NAME	ASCII (NAME)	SMITH	83	わたなべ XXX	33517	伊藤 XXX	35017						
NAME	ASCII (NAME)															
SMITH	83															
わたなべ XXX	33517															
伊藤 XXX	35017															
7	Q.	値の大小に関係なく、特定行の順序制御を行う方法はありますか。														
	A.	<p>ORDER BY 句を使用した順序制御では、照合順位に従って昇順、降順の並び替えが行われます。 しかし、ある行を最初や最後に表示したいといった、特定行に対する並び替えを行いたい場合は DECODE 関数と NULL の特徴を利用します。</p> <p><例> ENAME 列をもとに昇順の並び替えを行う際に、SCOTT さんの行を最初に表示する。 /* SCOTT を NULL と置き換え、ソート時に一番小さな値として認識させる */ SQL> select ename, deptno, empno from emp 2 where deptno = 20 3 order by decode(ename, 'SCOTT', null, ename) 4 nulls first;</p> <table border="1"> <thead> <tr> <th>ENAME</th> <th>DEPTNO</th> <th>EMPNO</th> </tr> </thead> <tbody> <tr> <td>SCOTT</td> <td>20</td> <td>7788</td> </tr> <tr> <td>ADAMS</td> <td>20</td> <td>7876</td> </tr> <tr> <td>FORD</td> <td>20</td> <td>7902</td> </tr> <tr> <td>…以下省略…</td> <td></td> <td></td> </tr> </tbody> </table>	ENAME	DEPTNO	EMPNO	SCOTT	20	7788	ADAMS	20	7876	FORD	20	7902	…以下省略…	
ENAME	DEPTNO	EMPNO														
SCOTT	20	7788														
ADAMS	20	7876														
FORD	20	7902														
…以下省略…																
8	Q.	グループ関数はどこまでネストすることができますか。														
	A.	<p>グループ関数は第 2 レベルまでネストを行うことができます。</p> <p><例>部門ごとの平均給与の最高額を表示する。 SQL> SELECT MAX(AVG(SAL)) FROM EMP GROUP BY JOB;</p>														

9	Q.	GROUP BY 句を使用する際の注意事項
	A.	<p>GROUP BY 句を使用する際は以下の点にご注意ください。</p> <ol style="list-style-type: none"> GROUP BY 句には別名を指定できない。 GROUP BY 句には副問合せを含めることができない。 式を含める場合は最大 255 個まで。 <p><例>GROUP BY 句に別名を指定しているため、エラーが発生する。 SQL> SELECT sum(sal), (select deptno from dept) AS demo FROM emp 2 GROUP BY demo; group by demo * 2 行でエラーが発生しました。 ORA-00904: 列名が無効です。</p>
10	Q.	日付関数に ROUND 関数を使用した場合の評価について。
	A.	<p>データを四捨五入する ROUND 関数を日付関数と一緒に使用した場合、現在の日付に最も近い午前 0 時の日付に四捨五入されます。</p> <p><例>現在、2003 年 7 月 20 日午後 7 時の場合 SQL> SELECT TO_CHAR(round(sysdate), 'YYYY/MM/DD HH24:MI:SS') AS days FROM dual;</p> <pre> DAYS ----- 2003/07/21 00:00:00 </pre>
11	Q.	DECODE 関数で、不等価な条件で値を比較する方法はありますか。
	A.	<p>DECODE 関数では不等価な条件での比較はできませんが、CASE 式を使用することで可能です。</p> <p><例> SELECT CASE WHEN 条件 THEN 処理 ←1 つ目の条件と合致した際の処理。 WHEN 条件 THEN 処理 ←2 つ目の条件と合致した際の処理。 ELSE 処理 ←どの条件にも合致しなかった際の処理。 END CASE ←CASE 式を終了。 FROM 表名;</p>
第 3 章 問合せの応用操作		
12	Q.	自然結合で同名の列が複数存在する場合、どういう条件で結合されるのでしょうか。
	A.	<p>同名の列全てが、等価結合の条件として実行されます。 例えば、2 つの表に A と B という同名の列が存在する場合、自然結合を行うと、以下の SQL と同等です。</p> <p><例> SELECT * FROM EMP JOIN DEPT ON EMP. A=DEPT. A ←同名 A 列の結合条件 AND EMP. B=DEPT. B; ←同名 B 列の結合条件</p>

13	Q.	結合条件に BETWEEN AND 演算子を使用した外部結合の方法
	A.	<p>結合条件に BETWEEN AND 演算子を指定した場合は BETWEEN 演算子の左辺、もしくは右辺に外部結合演算子(+)を指定します。 ※右辺には2列指定する事になりますが、(+)の指定は片方の列でも両方の列でも値を戻すことができます。</p> <p><例>EMP 表と SALGRADE 表の結合 /* EMP 表にしかないデータも結合の結果として表示する */ SQL> SELECT ENAME, SAL, GRADE FROM EMP, SALGRADE 2 WHERE SAL BETWEEN LOSAL(+) AND HISAL(+);</p> <p>/* SALGRADE 表にしかないデータも結合の結果として表示する */ SQL> SELECT ENAME, SAL, GRADE FROM EMP, SALGRADE 2 WHERE SAL(+) BETWEEN LOSAL AND HISAL;</p>
14	Q.	集合演算とインラインビューを使用した時に、インラインビュー内で ORDER BY 句を使用しているにも関わらず結果がソートされないのは何故ですか。
	A.	<p>結果がソートされることを保証できるのは、主問合せの式の最後に ORDER BY を使用したときのみです。そのため、以下のような場合は検索結果がソートされることが保障されません。</p> <p><例> SQL> SELECT ename, sal FROM (SELECT * FROM emp ORDER BY sal DESC) 2 WHERE ROWNUM <= 10 3 MINUS 4 SELECT ename, sal FROM (SELECT * FROM emp ORDER BY sal DESC) 5 WHERE ROWNUM <= 5;</p>
第5章 トランザクション制御		
15	Q.	DDL 操作で変更作業中の表に対し、処理を行うことはできますか。
	A.	<p>変更作業中の表に対して処理を行うことはできません。</p> <p>DDL 操作では、変更作業中にデータの食い違いが発生しないように、表に対してロックを取得します。(DDL、DML がロック待機の対象になります。)</p> <p>このロックは DDL 処理が終了するまで取得されます。</p>
第6章 DDL の基本操作		
16	Q.	スキーマ名の変更はできますか。
	A.	<p>SQL 文でスキーマ名の変更を行うことはできません。 スキーマ名の変更を行いたい場合はユーザーを再作成し、DataPump Export/Import を使用してデータ移行を行います。</p> <p>※ユーザーの作成方法については「データベース・マネジメント」コースでご紹介しています。 ※Data Pump Export/Import の詳細は「ユーティリティ」マニュアルをご参照ください。</p>

17	Q.	NUMBER 型の特殊な精度の指定方法。			
	A.	<p>NUMBER 型の特殊な精度の指定方法を、以下にご紹介します。</p> <ul style="list-style-type: none"> ◆ 正の値を表示させずに、小数点以下の桁数だけを指定する 全体の精度に「*」を指定します。 ◆ 特定の桁数で強制的に四捨五入する 小数点以下の精度指定に負の数値を指定すると、指定した桁数で四捨五入した値が挿入されます。 <p><例></p> <pre>SQL> CREATE TABLE test 2 (a NUMBER(*, 3), →小数点以下 3 桁のみを表示する 3 b NUMBER(5, -2)); →5 桁の数値を 2 桁目の数値で四捨五入して挿入する</pre> <pre>SQL> INSERT INTO test(a, b) 2 VALUES (0.387, 12345);</pre> <pre>SQL> SELECT * FROM test;</pre> <table style="margin-left: 40px; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">A</th> <th style="text-align: center;">B</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">.387</td> <td style="text-align: center;">12300</td> </tr> </tbody> </table>	A	B	.387
A	B				
.387	12300				
18	Q.	列のデフォルト値や仮想列についての情報を確認する方法。			
	A.	<p>列のデフォルト値や仮想列についての情報は USER_TAB_COLUMNS ビューの以下の列で確認することができます。</p> <ul style="list-style-type: none"> ・ DATA_DEFAULT 列 …列のデフォルト値（仮想列の場合は列式）が表示されます。 ・ VIRTUAL_COLUMN 列 …通常の場合は NO、仮想列の場合や YES と表示されます。 			
19	Q.	仮想列は、問合せ文の選択リストにしか指定できないのですか。			
	A.	<p>問合せ文のほか、DML 処理の WHERE 句の条件値としても使用できます。</p> <p>ただし、仮想列は実データを含みませんので、UPDATE の SET 句などでデータを直接操作することはできません。</p>			
20	Q.	複合主キー制約を定義する際の注意点。			
	A.	<p>表制約構文を使用することで複数列に一つの PRIMARY KEY 制約を定義することができます。</p> <p>ただし、一つの複合主キーは 33 個以上の列を持つことはできません。</p>			

21	Q.	副問合せを使用した表の作成 (AS SELECT での作成) を行う際にコピーされる NOT NULL 制約の名前はどのように定義されますか。											
	A.	<p>CONSTRAINT 句で制約名を指定しない場合と同じように Oracle が自動的に SYS_CXXX の形式で自動的に制約名を割り当てます。</p> <p>※制約名を分かりやすいものに変更したい場合は、ALTER TABLE ... RENAME 文を実行してください。</p>											
22	Q.	1つの CHECK 制約でユーザー任意の複数の条件を定義することは可能でしょうか。											
	A.	<p>CHECK 制約の条件の中で AND 論理演算子や OR 論理演算子を使用することで可能です。</p> <p><例>TEST 表の A 列に大文字であることと、文字数が 4 文字以上のデータのみを許可する</p> <pre>SQL> CREATE TABLE TEST 2 (A VARCHAR(10) CONSTRAINT CK_A CHECK(A = UPPER(A) AND LENGTH(A)>=4));</pre>											
23	Q.	CHECK 制約の内容を確認することはできますか。											
	A.	<p>USER_CONSTRAINTS ビューの SEARCH_CONDITION 列を確認してください。</p> <p>※NOT NULL 制約が定義されている場合、この列に「... IS NOT NULL」と表示されます。</p> <p><例>EMP 表に定義されている制約の情報を確認する。</p> <pre>SQL> SELECT table_name, constraint_name, constraint_type, search_condition 2 FROM user_constraints 3 WHERE table_name = 'EMP';</pre> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>TABLE_NAME</th> <th>CONSTRAINT_NAME</th> <th>C SEARCH_CONDITION</th> <th></th> </tr> </thead> <tbody> <tr> <td>EMP</td> <td>ENAME_NOTNULL</td> <td>C "ENAME" IS NOT NULL</td> <td>←NOT NULL 制約</td> </tr> <tr> <td>EMP</td> <td>ENAME_CK</td> <td>C ENAME&gt;LENGTH(3)</td> <td>←CHECK 制約</td> </tr> </tbody> </table>	TABLE_NAME	CONSTRAINT_NAME	C SEARCH_CONDITION		EMP	ENAME_NOTNULL	C "ENAME" IS NOT NULL	←NOT NULL 制約	EMP	ENAME_CK	C ENAME>LENGTH(3)
TABLE_NAME	CONSTRAINT_NAME	C SEARCH_CONDITION											
EMP	ENAME_NOTNULL	C "ENAME" IS NOT NULL	←NOT NULL 制約										
EMP	ENAME_CK	C ENAME>LENGTH(3)	←CHECK 制約										
24	Q.	既存データで制約に違反しているデータを特定する方法はありますか。											
	A.	<p>制約作成時に、exceptions オプションを指定することにより、制約に違反しているデータを EXCEPTIONS 表に取り出し特定することができます。</p> <p>※exceptions オプションの詳細については「SQL 言語リファレンス」マニュアルにてご参照ください。</p> <p>※EXCEPTIONS 表の詳細については、「リファレンス」マニュアルをご参照ください。</p>											
25	Q.	値の範囲をもとにして行うリレーションも FOREIGN KEY 制約によって保証することはできますか。											
	A.	<p>値の範囲をもとにして行うリレーションは、FOREIGN KEY 制約によって保証することはできません。FOREIGN KEY 制約は、参照先に同じ値が 1 つだけあることを保証する制約だからです。</p> <p>値の範囲をもとにして行うリレーションを保証したい場合は、トリガーを作成する必要があります。</p> <p>※トリガーの詳細については、「PL/SQL プログラミング I」コースでご紹介しています。</p>											

26	Q.	FOREIGN KEY 制約に定義したオプションのタイプを確認することはできますか。																			
	A.	<p>USER_CONSTRAINTS ビューの DELETE_RULE 列で、オプションの情報を確認することができます。</p> <p><例> SQL> SELECT table_name, constraint_name, constraint_type, delete_rule 2 FROM user_constraints 3 WHERE table_name = 'REFTEST';</p> <table border="1"> <thead> <tr> <th>TABLE_NAME</th> <th>CONSTRAINT_NAME</th> <th>C</th> <th>DELETE_RU</th> <th></th> </tr> </thead> <tbody> <tr> <td>REFTEST</td> <td>REF_KEY1</td> <td>R</td> <td>NO ACTION</td> <td>←オプションなし</td> </tr> <tr> <td>REFTEST</td> <td>REF_KEY2</td> <td>R</td> <td>CASCADE</td> <td>←「ON DELETE CASCADE」オプション付きの場合</td> </tr> <tr> <td>REFTEST</td> <td>REF_KEY3</td> <td>R</td> <td>SET NULL</td> <td>←「ON DELETE SET NULL」オプション付きの場合</td> </tr> </tbody> </table>	TABLE_NAME	CONSTRAINT_NAME	C	DELETE_RU		REFTEST	REF_KEY1	R	NO ACTION	←オプションなし	REFTEST	REF_KEY2	R	CASCADE	←「ON DELETE CASCADE」オプション付きの場合	REFTEST	REF_KEY3	R	SET NULL
TABLE_NAME	CONSTRAINT_NAME	C	DELETE_RU																		
REFTEST	REF_KEY1	R	NO ACTION	←オプションなし																	
REFTEST	REF_KEY2	R	CASCADE	←「ON DELETE CASCADE」オプション付きの場合																	
REFTEST	REF_KEY3	R	SET NULL	←「ON DELETE SET NULL」オプション付きの場合																	
27	Q.	FOREIGN KEY 制約を定義するときに、REFERENCES 句で指定する参照先の列名を省略できますか。																			
	A.	参照される表に PRIMARY KEY 制約が定義されていれば省略することができます。 (他にいくつ制約が定義されていても構いません)																			
28	Q.	ON DELETE CASCADE オプションによって削除された子表の行をロールバックする事はできますか。																			
	A.	<p>ロールバックすることができます。</p> <p>親表の行を削除する処理も、ON DELETE CASCADE オプションによって子表の行を削除する処理も同一トランザクション内で行われています。</p> <p>そのため、ロールバックコマンドを実行することで、親表の行を削除する処理と子表の行を削除する処理を取り消すことが出来ます。</p>																			
29	Q.	ON DELETE SET NULL オプションを使用して親表の行を削除しようとしたとき、外部キーに NOT NULL 制約が定義されているとどうなりますか。																			
	A.	ON DELETE SET NULL オプションによって削除する行に対応する外部キーの値を NULL に更新しようとしたますが、外部キーには NOT NULL 制約が定義されているためエラーとなります。																			
30	Q.	ON DELETE CASCADE、ON DELETE SET NULL オプションは表作成後に追加することはできますか。																			
	A.	FOREIGN KEY 制約を同時に定義する場合がありますが、ALTER TABLE ... ADD CONSTRAINT 文を使うことで、表作成後にオプションを追加することができます。																			
31	Q.	FOREIGN KEY 制約を ON DELETE CASCADE オプション付きで設定したときに、親表の行を TRUNCATE するとどのような動作になりますか。																			
	A.	<p>ON DELETE CASCADE オプションを使用して親表のデータを TRUNCATE することはできません。この場合、TRUNCATE 実行時にエラーが返ります。</p> <p>※Oracle 12c からは親表の TRUNCATE 実行時に CASCADE オプションを指定することで、子表のデータもまとめて削除できます。</p>																			

32	Q.	無効な FOREIGN KEY 制約を有効に戻す際に、参照キーに設定された PRIMARY KEY 制約や UNIQUE KEY 制約を同時に有効にするオプションはありますか。
	A.	FOREIGN KEY 制約と、参照キーに設定された制約を同時に有効にすることはできません。各制約を一つずつ有効に変更する必要があります。 ※ALTER TABLE ... DISABLE CONSTRAINT 文で CASCADE オプションを使用することで、各制約を同時に無効にすることはできます。
33	Q.	FOREIGN KEY 制約を定義して親表、子表、孫表という関係にある時に、親表の行を削除すると対応する子表、孫表の行まで削除させる事はできますか。
	A.	子表、孫表に定義する FOREIGN KEY 制約をに対して、それぞれオプション (ON DELETE CASCADE、ON DELETE SET NULL) を指定することで実現可能です。
34	Q.	表やビューなどにコメントなどの説明文を加えることはできますか。
	A.	COMMENT コマンドを使用することで可能です。 定義した内容は USER_TAB_COMMENTS ビュー(表に対するコメント)、または USER_COL_COMMENTS(列に対するコメント)で確認します。 <例> COMMENT ON TABLE 表名 IS 'コメント'; ←表に対してコメントを加える。 SELECT * FROM USER_TAB_COMMENTS; ←表に対するコメントを確認する。 COMMENT ON COLUMN 表名.列名 IS 'コメント'; ←列に対してコメントを加える。 SELECT * FROM USER_COL_COMMENTS; ←列に対するコメントを確認する。
35	Q.	*(アスタリスク)を使用してビューを作成した場合、ビュー作成後に実表に追加した列はビューに反映されますか。
	A.	反映されません。*を使用してビューを作成した場合、*は実際の列名に展開されてデータ・ディクショナリに登録されます。 ちなみに新しく追加した列をビューに反映させたい場合はビューを再作成する必要があります。
36	Q.	読取り専用のビューを作成する方法。
	A.	ビューの定義に WITH READ ONLY オプションを指定することで、読み取り専用のビューを作成することができます。この読み取り専用ビューに対して DML 操作を行うとエラーとなります。 <例>VIEW_EMP 表を読取り専用で作成する。 SQL> CREATE VIEW view_emp AS SELECT ename FROM emp WITH READ ONLY;

37	Q.	表内の特定列のみに UPDATE 権限を付与する方法。
	A.	<p>UPDATE 権限に限って、特定列に権限を付与することができます。</p> <p>特定列のみに UPDATE 権限を付与する場合は、UPDATE 権限の後に権限を付与する列を記述します。</p> <p><例>EMP 表の ENAME、SAL、COMM 列に対する UPDATE 権限を SCOTT ユーザーに付与する SQL> GRANT update(ename, sal, comm) ON emp TO scott;</p> <p>UPDATE コマンドは列単位の処理であり、UPDATE コマンドの後のカッコ内に列を記述することで指定された列のみへの処理を許可します。</p>
38	Q.	同じ権限の組み合わせを一つにまとめて管理することはできますか。
	A.	<p>ロールを使用すれば、複数の権限を一つにまとめて管理することができます。</p> <p>ロールとは関連する権限のグループに名前をつけたもので、ロールをユーザーに付与する事でロール内の権限を一括して付与することができます。</p> <p>※ロールについての詳細は「データベース・マネジメント」コースでご紹介しています。</p>
付録		
39	Q.	小数点以下の数値を表示させる方法。
	A.	<p>TO_CHAR 関数を使用することで小数点以下のデータを表示させることができます。 ※SQL*Plus を使用している場合は、COLUMN コマンドを使うこともできます。</p> <p><例>sal 列を表示する際、小数第 2 位まで表示する /* TO_CHAR 関数を使った場合 */ SQL> SELECT TO_CHAR(sal, '999.99') FROM emp;</p> <p>/* COLUMN コマンドを使った場合 */ SQL> COLUMN sal FORMAT 999.99</p>
40	Q.	日付データを和暦で表示することはできますか。
	A.	<p>初期化パラメータ NLS_CALENDAR を JAPANESE IMPERIAL に変更することで和暦で表示できます。</p> <p><例>現在のセッションで和暦を表示させる SQL> ALTER SESSION SET NLS_CALENDAR = 'JAPANESE IMPERIAL';</p> <p>SQL> SELECT sysdate FROM dual;</p> <p>SYSDATE ----- 平成 21 年 08 月 06 日</p> <p>※初期化パラメータ NLS_CALENDAR の詳細については「リファレンス」マニュアルをご参照ください。</p>

41	Q.	FLASHBACK TABLE による表の復元が行える期間について。
	A.	<p>Oracle 10g R1 以降は、PURGE オプションを使用せずにテーブルを削除した場合、その表は物理的には残っています。そのため、FLASHBACK TABLE 文によって復元することができます。</p> <p>しかし、削除された表が使っている物理的な領域は、削除されていない他の表や、新規に作成される表に優先的に使用されていきます。</p> <p>そのため、削除された表の領域が他のオブジェクトによって使用されてしまうと表の復元が行えなくなります。つまり、他のオブジェクトに領域が再利用されるまでが、表の復元を行える期間となります。</p>
42	Q.	DUAL 表にアクセスをするときにスキーマ名を修飾が不要な理由
	A.	<p>DUAL 表にはシノニムという、オブジェクトに対する別名が設定されているためです。</p> <p>シノニムは表などのデータベースオブジェクトに設定する別名のことをいい、以下のような目的で使用します。</p> <ul style="list-style-type: none"> ・オブジェクトの名前と所有者を隠蔽化する。 ・ユーザー側の SQL 文を単純なものとし、アクセスの簡素化を実現。 <p>※シノニムの詳細については「概要」マニュアルをご参照ください。</p>
43	Q.	紀元前の日付情報を扱うにはどうすればよいでしょうか。
	A.	<p>TO_DATE 関数や TO_CHAR 関数などを使用して、日付書式を指定します。</p> <p><例>紀元前の年号を負の値で表示する。(日付書式「s」を使用する) SQL>SELECT TO_CHAR(hiredate, 'syyyy') FROM emp;</p> <p><例>紀元前 4712 年のデータ挿入する。(日付書式 BC、AD を使用する) ※BC には「紀元前」、AD には「西暦」という日本語を使用します。</p> <p>SQL>INSERT INTO tab_date(col_date) 2 VALUES(to_date('紀元前 47120101', 'bcyyyymmdd'));</p>
44	Q.	日付データの曜日を英語で表示することはできますか。
	A.	<p>初期化パラメータ NLS_DATE_LANGUAGE を JAPANESE から ENGLISH に変更することで可能です。以下はセッションレベルでパラメータを変更した例です。</p> <p><例>現在のセッションで日付データを英語で表示させる SQL>ALTER SESSION SET NLS_DATE_LANGUAGE = 'ENGLISH';</p> <p>※初期化パラメータ NLS_DATE_LANGUAGE の詳細については「リファレンス」マニュアルをご参照ください。</p>

45	Q.	デフォルトの日付書式を変更することはできますか。
	A.	<p>ALTER SESSION 文を使用して、デフォルトの日付書式を変更できます。</p> <p><例>デフォルトの日付書式を変更する。 SQL> ALTER SESSION 2 SET NLS_DATE_FORMAT=' YYYY/MM/DD HH24:MI' ;</p>
46	Q.	西暦の上 2 桁だけ評価できる日付書式はありますか。
	A.	<p>西暦の上 2 桁だけを評価できる日付書式はありません。 代替案としては SUBSTR 関数を使用し、評価したい 2 桁分だけを抜き出す方法が挙げられます。</p> <p><例> SQL> SELECT SUBSTR(TO_CHAR(hiredate, 'yyyy'), 1, 2) FROM emp;</p> <p>SUBS ----- 19 …以下省略…</p>
その他		
47	Q.	' (シングルコーテーション) を文字列として認識させるにはどのように指定したら良いですか。
	A.	<p>文字列に挿入したいシングルコーテーションの直前にシングルコーテーションをひとつ指定することで、条件値を指定するシンボルではなく、文字データとして Oracle に認識させる事ができます。</p> <p><例> /* EMP 表の ENAME 列に TOM' というデータを挿入する */ SQL> INSERT INTO emp(empno, ename) 2 VALUES(1122, ' TOM' ');</p> <p>/* 挿入したデータを確認する */ SQL> SELECT empno, ename FROM emp 2 WHERE ename = ' TOM' ' ;</p> <p>EMPNO ENAME ----- 1122 TOM'</p>
48	Q.	SQL*Plus コマンドの EDIT コマンドで起動するエディタを変更することはできますか。
	A.	<p>_EDITOR 変数を設定することで、起動するエディタを変更することができます。 以下のように変数に対して、使用したいエディタの EXE ファイルのパスを指定します。</p> <p><例> DEFINE _EDITOR = "C:\Program Files\EmEditor\EmEditor.exe"</p>

49	Q.	SQL*Plus で問合せ結果を表示した際に、点線を表示させない方法。
	A.	<p>SQL*Plus で通常問合せを行うと、列ヘッダと行データの間点線が表示されます。この点線を表示させたくない場合は SQL*Plus のシステム変数 UNDERLINE を OFF に設定してください。</p> <p><例> SQL> SET UNDERLINE OFF SQL> SELECT * FROM DEPT1;</p> <pre> DEPTNO DNAME LOC 10 ACCOUNTING NEW YORK </pre>
50	Q.	SQL*Plus で問合せ実行時に結果が何行返されたのか確認する方法
	A.	<p>SQL*Plus のシステム変数 FEED を ON に設定してください。また逆に結果を表示させたくない場合は OFF に設定してください。</p> <p><例> SQL> SET FEED OFF SQL> SELECT ename FROM emp;</p> <pre> ENAME ----- SMITH ...省略... MILLER SQL> </pre>
51	Q.	SQL*Plus を使用した環境で、数値データが指数表示 (A × (B の C 乗)) されてしまうのはなぜですか。
	A.	<p>SQL*Plus 側で数値を表示する列幅が実データの桁数に比べて小さいため発生する問題であるため、SQL*Plus の列幅表示を拡張すると解消されます。</p> <p>列幅は、以下の方法で拡張します。</p> <ol style="list-style-type: none"> 1. COLUMN コマンド (特定列を調整する) COLUMN 列名 FORMAT 書式 2. NUMFORMAT システム変数 (表の数値列全てを調整する) SET NUMFORMAT 書式 3. NUMWIDTH システム変数 (表の数値列全てを調整する) SET NUMWIDTH 桁数
52	Q.	SQL*Plus の環境設定コマンドの値をはじめから有効にしておくことはできますか。
	A.	<p>SQL*Plus 起動時に自動的に実行される glogin.sql ファイルを使用します。この glogin.sql ファイル内に環境設定コマンドを記述しておくことで、SQL*Plus 起動時に定義が有効になります。</p> <p>※ファイルの作成場所 \$ORACLE_HOME/sqlplus/admin</p>

53	Q.	SQL*Plus で OS コマンドを実行する方法
	A.	HOST コマンドを使用すると、SQL*Plus で OS コマンド (Windows、Linux) を実行できます。
54	Q.	ユーザーがオブジェクトに対して最後に変更処理した日付を取得できますか。
	A.	DDL 文による変更作業は USER_OBJECTS ビューの LAST_DDL_TIME 列で確認できます。 DML 文における変更作業はデータ・ディクショナリで管理していないため、データ・ディクショナリから確認を行うことはできませんが、以下の機能を使用することで可能になります。 1. 監査機能を利用する。 ※Oracle の監査機能については、「データベース・マネジメント」コースでご紹介しています。 2. トリガーを利用する。 ※トリガーとは表に対する処理をきっかけに、暗黙的に起動するプログラムです。 ※トリガーの詳細については「PL/SQL プログラミング I」コースでご紹介しています。
55	Q.	SQL*Plus で「&」を文字データとして認識させることはできますか。
	A.	「&」（置換文字）のように SQL*Plus 上で特殊な意味をもつ記号を文字データとして扱うためには、SQL*Plus の ESCAPE コマンドを使用することで可能です。以下に例をご紹介します。 SET ESCAPE ¥ INSERT INTO test values('¥&'); ※上記の ESCAPE コマンドはあくまで SQL*Plus のコマンドであり、SQL の LIKE 演算子で使用する ESCAPE コマンドとは別のものになりますのでご注意ください。

※ ご利用上の注意事項※

- ・本書の著作権は株式会社アシストに帰属します。
- ・本書は参考資料であり、掲載されている情報は予告なしに変更されることがあります。
- ・本書で使用している製品の名称は、各社の商標または登録商標です。
- ・本資料の内容に関するご質問はご遠慮ください。
- ・本資料はお客様の責任のもとでご利用ください。これらの使用によりいかなる損害が生じたとしても、株式会社アシストは一切保証致しかねますので、ご了承ください。