

連番	ページと該当箇所	誤		正	
		赤字：修正箇所（修正前）	緑字：説明その他	青字：修正箇所（修正後）	緑字：説明その他
1	25ページ 「EMPLOYEES表」データ5行目	EMAIL EMAIL NOT NULL VARCHAR2(25) Eメールアドレス		1行追加： EMAIL EMAIL NOT NULL VARCHAR2(25) Eメールアドレス PHONE_NUMBER VARCHAR2(20) 電話番号	
2	26ページ CUSTOMERS表：データ最下行 および COUNTRIES表：データ1行目	COUNTRY_ID NOT NULL NUMBER 国ID		COUNTRY_ID NOT NULL CHAR(2) 国ID	
3	35ページ 11行目 （「参考」3行目）	IFED1XやCASE*METHODがあります。		IDEF1XやCASE*METHODがあります。	
4	47ページ 8行目 （「NULLとは」本文7行目）	それは、定義されていないデータに対して		それは、定義されていないデータに対して	
5	52ページ 7行目	SELECT last_name, 'commission is ' last_name		SELECT last_name, 'commission is ' commission_pct	
6	63ページ 下から3～2行目	指定してます。		指定しています。	
7	75ページ 下から5行目 （「解答への道すじ」最下行）	WHERE job_id LIKE 'SA¥%' ESCAPE '¥';		WHERE job_id LIKE 'IT¥%' ESCAPE '¥';	
8	80ページ 8行目	WHERE depaertment_id N (30,40,60);		WHERE depaertment_id IN (30,40,60);	
9	81ページ下から11行目 82ページ4行目 （81ページ「使用例」リスト1 行目および82ページ最初のリス ト1行目）	SELECT employee_id, last_name, salary, department_id		SELECT employee_id, last_name, department_id	
10	85ページ 3～4行目 （「例題」本文2～3行目）	EMPLOYEE_IDとLAST_NAME		EMPLOYEE_ID、LAST_NAME、SALARY、DEPARTMENT_ID	
11	88ページ 下から4行目 （「ソートの例4」説明文3行	値の小さい順		ANNSAL列の小さい順	
12	91ページ 5～6行目	条件や列名などを一部を変更して		条件や列名などを一部変更して	
13	98ページ 図3-1	(1) 図中の中央下（注意書きの上）付近の「Grant」 (2) 左下の四角い枠内のGrant (3) 図中の右側中央付近の文字 指定した関数を解釈して、表内のデータを表示のために加工		(1) 「GRANT」 (2) GRANT (3) 指定した関数を解釈し、表内のデータを取り出して表示のために加工 （この図の例では大文字に変換）	
14	105ページ 下から2行目	また、第3引数を省略したときは第1引数に指定した～		また、第3引数を省略したときは第2引数に指定した～	

連番	ページと該当箇所	誤 赤字：修正箇所（修正前） 緑字：説明その他	正 青字：修正箇所（修正後） 緑字：説明その他
15	107ページ 7～9行目 (INSTR関数の書式、記述法)	書式 INSTR(列名 式, '文字列'[, m][, n]) 記述法 関数が受け取る引数に列名または文字列を指定します。後の引数には数値を指定します。	書式 INSTR(列名 式, '文字列'[, m[, n]]) 記述法 関数が受け取る引数に、列名または文字列、 検索文字列 を指定します。後の引数には数値を指定します。mは検索開始位置、nは出現数で、m以降に出現した文字列のn番目の出現位置を戻します。m、nのデフォ
16	107ページ 下から6～3行目 (「LPAD関数」の書式および記述法)	書式 ----- LPAD(列名 式, n, '文字列') ----- 記述法 関数が受け取る引数に列名または文字列を指定します。後の引数には数値、 文字列 の順に指定します。	書式 ----- LPAD(列名 式, n, '文字') ----- 記述法 関数が受け取る引数に列名または文字列を指定します。後の引数には数値、 文字 の順に指定します。
17	108ページ 下から13～10行目 (RPAD関数の書式および記述法)	書式 ----- RPAD(列名 式, n, '文字列') ----- 記述法 関数が受け取る引数に列名または文字列を指定します。後の引数には数値、 文字列 の順に指定します。	書式 ----- RPAD(列名 式, n, '文字') ----- 記述法 関数が受け取る引数に列名または文字列を指定します。後の引数には数値、 文字 の順に指定します。
18	109ページ 8～18行目 (TRIM関数の書式、記述法、機能)	書式 ----- TRIM([LEADING TRAILING BOTH] ['切り捨てる文字列' FROM] '切捨て対象の文字列') ----- 記述法 []内は任意で指定します。LEADING, TRAILING, BOTHのいずれも指定しない場合、「BOTH」で結果を戻します。切り捨て対象の文字列の左右両方向から、 切り捨てる文字列 を切り捨てることができます。また、片方の文字を切り捨てる時は、明示的に「LEADING（先頭文字）」「TRAILING（末尾文字）」を指定します。関数が受け取る引数に列名または文字列を指定します。 機能 文字列の先頭もしくは末尾から指定した文字を切り捨てます。また、 切り捨てる文字列 を省略した場合、半角スペースが切り捨て対象です。	書式 ----- TRIM([LEADING TRAILING BOTH] ['切り捨てる文字' FROM] '切捨て対象の文字列') ----- 記述法 []内は任意で指定します。LEADING, TRAILING, BOTHのいずれも指定しない場合、「BOTH」で結果を戻します。切り捨て対象の文字列の左右両方向から、 指定した文字 を切り捨てることができます。また、片方の文字を切り捨てる時は、明示的に「LEADING（先頭文字）」「TRAILING（末尾文字）」を指定します。関数が受け取る引数に列名または文字列を指定します。 機能 文字列の先頭もしくは末尾から指定した文字を切り捨てます。また、 切り捨てる文字 を省略した場合、半角スペースが切り捨て対象です。
90	110ページ 11行目 (REPLACE関数の書式)	REPLACE('置換対象の文字列', '置換前の文字列', ['置換後の文字列'])	REPLACE('置換対象の文字列', '置換前の文字列'[, '置換後の文字列'])
19	112ページ 下から6行目 (「使用例1」の説明文)	使用例1 小数点以下1桁を四捨五入します。	使用例1 小数点以下第二位を四捨五入し、第一位まで表示します。
20	113ページ 1～2行目 (「使用例2」の説明文)	使用例2 第2引数を省略するか、第2引数に0を指定すると、 整数の位（1の位） で四捨五入します。	使用例2 第2引数を省略するか、第2引数に0を指定すると、 小数点以下第一位 を四捨五入し、 整数の一の位 まで表示します。
21	113ページ 下から6行目 (「使用例1」の説明文)	使用例1 小数点以下1桁を切り捨てます。	使用例1 小数点以下第二位を切り捨てて、第一位まで表示します。
22	114ページ 1～2行目 (「使用例2」の説明文)	使用例2 第2引数を省略するか、第2引数に0を指定すると、 整数の位（1の位） で切り捨てます。	使用例2 第2引数を省略するか、第2引数に0を指定すると、 小数点以下第一位 を切り捨てて、 整数の一の位 まで表示します。
23	115ページ 6行目 (「日付関数」本文2行目)	日付を世紀、年、月、日、時、分、秒を	日付の世紀、年、月、日、時、分、秒を

連番	ページと該当箇所	誤 赤字：修正箇所（修正前） 緑字：説明その他	正 青字：修正箇所（修正後） 緑字：説明その他
24	119ページ 15行目 （「使用例1」の説明）	2009年8月15日と2009年3月15日に月数を	2009年8月15日と2009年5月15日の間の月数を
25	122ページ 4～7行目 （ROUND関数の「記述法」「機能」「使用例」）	記述法 第1引数には日付値を指定します。第2引数は任意で文字列を指定します。 機能 第1引数に指定した日付を第2引数未満で四捨五入します。 使用例 2009年7月20日を月で四捨五入して表示します。	記述法 第1引数には日付値を指定します。第2引数には「結果としての単位まで結果を戻すか」を指定するための文字列（'YEAR'、'MONTH'など）を指定します（省略時は日単位）。 機能 第1引数に指定した日付を、第2引数で指定された単位未満の単位で四捨五入します。 使用例 現在の日付（2009年7月20日）を、月の単位まで戻すように四捨五入して表示します。
26	122ページ 下から9～6行目 （TRUNC関数の「記述法」「機能」「使用例」）	記述法 第1引数には日付値を指定します。第2引数は任意で文字列を指定します。 機能 第1引数に指定した日付を第2引数未満で切り捨てます。 使用例 2009年7月20日を月で切り捨てて表示します。	記述法 第1引数には日付値を指定します。第2引数には切り捨ての単位となる文字列（'YEAR'、'MONTH'など）を指定します（省略時は日単位）。 機能 第1引数に指定した日付を第2引数の単位で切り捨てます。 使用例 現在の日付（2009年7月20日）を月の単位で切り捨てて表示します。
27	139ページ 下から8～4行目	CASE 式 WHEN 条件1 THEN 値1 [WHEN 条件2 THEN 値2] [WHEN 条件3 THEN 値3] : [ELSE デフォルト値]	CASE 式 WHEN 条件1 THEN 値1 [WHEN 条件2 THEN 値2] [WHEN 条件3 THEN 値3] : [ELSE デフォルト値] END
28	140ページ 下から4～1行目 （DECODE関数「書式」）	(DECODE 列名 式, 条件1, 値1, [条件2, 値2,] [条件3, 値3,] : [ELSE デフォルト値])	DECODE (列名 式, 条件1, 値1, [条件2, 値2,] [条件3, 値3,] : [デフォルト値])
29	146ページ 7行目 （「グループ関数のガイドライン」本文6行目）	NVL関数をして、	NVL関数を使用して、
30	147ページ 下から8行目 （「グループ関数の使用例2」説明文1行目）	従業員の姓を	FIRST_NAMEを
31	150ページ 「使用例2」のリスト	ソートの指定がないのにソートされている	リストの右脇に赤字の注釈を追加： データのソート順は、説明を理解しやすいように適宜入れ替えている
32	151ページ 「複数列での検索例」のリスト	ソートの指定がないのにソートされている	リストの右脇に赤字の注釈を追加： データのソート順は、説明を理解しやすいように適宜入れ替えている
33	155ページ 最下行 （「解答」(3)最下行）	FROM employees HAVING AVG(sal) >= 2000	FROM employees HAVING AVG(salary) >= 2000

連番	ページと該当箇所	誤		正	
		赤字：修正箇所（修正前） 緑字：説明その他		青字：修正箇所（修正後） 緑字：説明その他	
34	163ページ 下から8～1行目 (リスト中の<実行結果>)	LAST_NAME ----- Fay Gietz Kochhar De Haan Ernst Austin	JOB_ID ----- MK_REP AC_ACCOUNT AD_VP AD_VP IT_PROG IT_PROG	DEPARTMENT_NAME ----- Marketing Accounting Executive Executive IT IT	EMPLOYEE_ID LAST_NAME DEPARTMENT_NAME ----- 202 Fay Marketing 206 Gietz Accounting 101 Kochhar Executive 102 De Haan Executive 104 Ernst IT 105 Austin IT 106 Pataballa IT
35	165ページ 下から3行目	FROM 表名1 [表別名1] JOIN [表名2 [表別名2]]		FROM 表名1 [表別名1] JOIN 表名2 [表別名2]	
36	166ページ 下から12行目	USING department_id;		USING (department_id);	
37	169ページ 3～4行目、12行目	PRODUCT表		PRODUCTS表	
38	171ページ 11行目 (「等価結合の例」リスト1行目)	SELECT employee_id, last_name, department_name, city		SELECT employee_id, last_name, department_name	
39	172ページ 図6-6	(ステップ1で結合した表のLOC_ID列のデータ) 1400 1500 1600 1700 1800 (ステップ2で結合した表のCITY列のデータ) Southlake South San Francisco South Brunswick Seattle Toronto		(ステップ1で結合した表のLOC_ID列のデータ) 1400 1400 1700 1800 1800 (ステップ2で結合した表のCITY列のデータ) Southlake Southlake Seattle Toronto Toronto	
40	173ページ 2行目 (「3つの表の結合例」説明文1行目)	従業員名、職種ID、部門名、所在地を		EMPLOYEE_ID、LAST_NAME、DEPARTMENT_NAME、CITYを	
41	174ページ 9行目 (「ここに着眼！」本文2行)	PRODUCT表		PRODUCTS表	
42	174ページ 14行目 (「解答への道すじ」本文1行目)	CUST_FIRAT_NAME		CUST_FIRST_NAME	
43	177ページ 図6-8 右上の表の名前	JOB_GRADE表		JOB_GRADES表	
44	177ページ 図6-8内で、結合した表のGRADE列のデータ	A B C D E		A A B A C	

連番	ページと該当箇所	誤		正	
		赤字：修正箇所（修正前）	緑字：説明その他	青字：修正箇所（修正後）	緑字：説明その他
45	178ページ 下から5行目 （「非等価結合の例2」リスト3 行目）	j.highest_sal		j.highest_sal;	
46	179ページ 表6-2内の「結合方法（句）」	RIGHT OUTOER JOIN		RIGHT OUTER JOIN	
47	181ページ 下から12行目 （下のリストの2行目）	FROM employees e LEFT OUTER JOIN departments d （下線は背景赤色の範囲です）		FROM employees e LEFT OUTER JOIN departments d （下線は背景赤色の範囲です）	
48	183ページ 図6-11 上端のSQL文2行目	FROM employees e LEFT OUTER JOIN departments d		FROM employees e FULL OUTER JOIN departments d	
49	184ページ 6行目 （リストの2行目）	FROM employees e FULL OUTER JOIN departments d （下線は背景赤色の範囲です）		FROM employees e FULL OUTER JOIN departments d （下線は背景赤色の範囲です）	
50	185ページ 「例題」全体	（p174の「例題」と同じものが重複して入っていました）		（例題を削除）	
51	186ページ 5行目 （「クロス結合」本文3行目）	指定した表と結合 の する表のすべての行		指定した表と結合 する 表のすべての行	
52	186ページ 図6-12内の下の表	----- EMP_ID L_NAME DEPT_ID D_NAME ----- 198 Oconnell 10 Administration 199 Grant 10 Administration 200 Whalen 10 Administration 201 Hartstein 10 Marketing 202 Fay 10 marketing 198 Oconnell 20 marketing 199 Grant 20 marketing 200 Whalen 20 marketing 201 Hartstein 20 Marketing 202 Fay 20 marketing -----		----- EMP_ID L_NAME D_NAME ----- 198 Oconnell Administration 199 Grant Administration 200 Whalen Administration 201 Hartstein Administration 202 Fay Administration 198 Oconnell marketing 199 Grant marketing 200 Whalen marketing 201 Hartstein Marketing 202 Fay marketing -----	
53	188ページ 6行目 （確認問題(1)）	記述するときの注意点を		記述 する ときの注意点を	
54	188ページ 9行目、下から3行目 （確認問題(3)と解答(3)）	「確認問題」 (3) 「表は 2つ 以上結合できない」この内容は正しいか誤りか述べてよ 「解答」 (3) 誤り。結合条件が あれば2つ 以上の表を結合できる		「確認問題」 (3) 「表は 3つ 以上結合できない」この内容は正しいか誤りか述べてよ 「解答」 (3) 誤り。結合条件が 揃えば3つ 以上の表を結合できる	
55	196ページ 7行目 （本文3～4行目）	主問合せに指定できる比較演算子のことを複数行比較演算子 ） といい、		主問合せに指定できる比較演算子のことを複数行比較演算子 といい、	

連番	ページと該当箇所	誤 赤字：修正箇所（修正前） 緑字：説明その他	正 青字：修正箇所（修正後） 緑字：説明その他
56	196ページ 表7-2内で、下から2～1行目	ANY 値を、内側の問合せから戻された各値と比較 ALL 値を、内側の問合せから戻されたすべての値と比較	ANY 副問合せから戻されたすべての値と比較し、1つ以上の比較が成り立った場合に、全体として条件が成り立つ ALL 副問合せから戻されたすべての値と比較し、すべての比較が成り立った場合に、全体として条件が成り立つ
57	198ページ 8～9行目 （「ANY演算子の使用例1」説明1～2行目）	DEPARTMENT_IDが60である従業員のSALARYより多くもらっている従業員の情報を表示します。	DEPARTMENT_IDが60である従業員のSALARY列のいずれかより少ない（最大値より小さい）SALARYの従業員について情報を表示します。
58	200ページ 4～5行目 （「ALL演算子の使用例」説明1～2行目）	DEPARTMENT_IDが60である従業員のSALARYより多くもらっている従業員の情報を表示します。	DEPARTMENT_IDが60である従業員のSALARY列のどれよりも少ない（最小値より小さい）SALARYの従業員について情報を表示します。
59	201ページ 2～3行目	ANY演算子と「!=」と組み合わせ、	ANY演算子と「!=」の組合せ、
60	202ページ 16行目 （「1つ目の問合せ」のSQL	SELECT department_id FROM departments	SELECT department_id FROM departments WHERE department_name IN ('Sales', 'Marketing');
61	204ページ 11行目 （項見出し）	単一行関数	単一行副問合せ
62	206ページ 下から4行目からのリスト全部	SELECT last_name, job_id FROM employees WHERE employee_id != (SELECT manager_id FROM employees WHERE manager_id = 100 AND manager_id = 123 AND manager_id = NULL);	SELECT last_name, job_id FROM employees WHERE employee_id != 100 AND employee_id != 123 AND employee_id != NULL;
63	207ページ 下から2～1行目 （「まとめ」の内容）	内側の問合せから戻される値がない場合は、NULLと比較していることと同じであるため、結果を戻さない	主問合せでNOT INを使用している場合、副問合せから戻された結果の中にNULLが1行でも含まれていると、主問合せは結果を戻さない
64	208ページ 13行目	(SELECT mgr_id FROM employees);	(SELECT manager_id FROM employees);
65	208ページ 下から3～1行目	または WHERE employee_id IN (SELECT manager_id FROM employees);	この3行を削除
66	214ページ 12行目 （本文2行目）	MANAGERS表にはSALARY列がありません。	DEPARTMENTS表にはSALARY列がありません。
67	217ページ 下から12～10行目 （リスト1～3行目）	SELECT employee_id, last_name FROM employees INTERSECT SELECT employee_id, last_name FROM managers;	SELECT employee_id, last_name FROM employees MINUS SELECT employee_id, last_name FROM managers;

連番	ページと該当箇所	<p style="text-align: center;">誤</p> <p style="text-align: center;">赤字：修正箇所（修正前） 緑字：説明その他</p>	<p style="text-align: center;">正</p> <p style="text-align: center;">青字：修正箇所（修正後） 緑字：説明その他</p>
68	256ページ 3行目以下（リスト全体） （「SELECT FOR UPDATE文の使用例」リスト）	<pre>SELECT employee_id, last_name FROM employees FOR UPDATE;</pre> <p><ユーザーA> EMPLOYEE_ID LAST_NAME ----- 198 OConnell 199 Grant : 197 Feeney</p> <pre>UPDATE employees SET employee_id = 1000 WHERE employee_id = 197;</pre> <p>1行が更新されました。</p> <p><ユーザーB> UPDATE employees SET employee_id = 2000 WHERE employee_id = 197; --待ち状態</p> <p><ユーザーA> UPDATE employees SET employee_id = 1000 WHERE employee_id = 197;</p> <p>1行が更新されました。</p> <pre>COMMIT;</pre> <p><ユーザーB> UPDATE employees SET employee_id = 2000 WHERE employee_id = 197;</p> <p>1行が更新されました。</p>	<p><ユーザーA> SELECT employee_id, last_name FROM employees FOR UPDATE;</p> <p>EMPLOYEE_ID LAST_NAME ----- 198 OConnell 199 Grant : 197 Feeney</p> <pre>UPDATE employees SET employee_id = 1000 WHERE employee_id = 197;</pre> <p>1行が更新されました。</p> <p><ユーザーB> UPDATE employees SET employee_id = 2000 WHERE employee_id = 197; --待ち状態</p> <p><ユーザーA> COMMIT;</p> <p><ユーザーB> 1行が更新されました。</p>
69	258ページ 図9-11内で、TRUNCATE文の説明	TRUNCATE文でデータを削除した場合は、領域が 残る	TRUNCATE文でデータを削除した場合は、領域が 残らない
70	261ページ 下から5行目 （「解答」(1) 2行目）	（例）VALUES (scott , NULL)	（例）VALUES (' Scott ', NULL)
71	266ページ 1行目	ネーミング規則に 即 する	ネーミング規則に 則 する
72	269ページ 図10-3の中の文字	VARCHAR2	VARCHAR2(5)

連番	ページと該当箇所	誤 赤字：修正箇所（修正前） 緑字：説明その他	正 青字：修正箇所（修正後） 緑字：説明その他																																																								
73	278ページ12～26行目 279ページ11～12行目 （「例題」）	<p>「例題」本文11～25行目</p> <p>デフォルト値： JOB_ID列に「IT_PROG」を設定</p> <p>表の作成後、挿入するデータ：</p> <table border="1"> <thead> <tr> <th>ID</th> <th>NAME</th> <th>JOB_ID</th> <th>OM_LEVEL</th> </tr> </thead> <tbody> <tr><td>1</td><td>ben</td><td>it_prog</td><td>silver</td></tr> <tr><td>2</td><td>mary</td><td>default</td><td>silver</td></tr> <tr><td>3</td><td>tom</td><td>sa_rep</td><td>gold</td></tr> </tbody> </table> <p>データ挿入後、デフォルト値の変更： JOB_ID列のデフォルト値「IT_PROG」を「IT_INFRA」に変更</p> <p>デフォルト値の変更後挿入するデータ</p> <table border="1"> <thead> <tr> <th>ID</th> <th>NAME</th> <th>JOB_ID</th> <th>OM_LEVEL</th> </tr> </thead> <tbody> <tr><td>4</td><td>bob</td><td></td><td>bronze</td></tr> <tr><td>5</td><td>andy</td><td>sa_rep</td><td>silver</td></tr> </tbody> </table> <p>「解答」本文10～11行目</p> <pre>INSERT INTO om_master VALUES(2, 'mary', default, 'silver');</pre>	ID	NAME	JOB_ID	OM_LEVEL	1	ben	it_prog	silver	2	mary	default	silver	3	tom	sa_rep	gold	ID	NAME	JOB_ID	OM_LEVEL	4	bob		bronze	5	andy	sa_rep	silver	<p>「例題」本文11～25行目</p> <p>デフォルト値： JOB_ID列に「it_prog」を設定</p> <p>表の作成後、挿入するデータ：</p> <table border="1"> <thead> <tr> <th>ID</th> <th>NAME</th> <th>JOB_ID</th> <th>OM_LEVEL</th> </tr> </thead> <tbody> <tr><td>1</td><td>ben</td><td>it_prog</td><td>silver</td></tr> <tr><td>2</td><td>mary</td><td>デフォルト値</td><td>silver</td></tr> <tr><td>3</td><td>tom</td><td>sa_rep</td><td>gold</td></tr> </tbody> </table> <p>データ挿入後、デフォルト値の変更： JOB_ID列のデフォルト値「it_prog」を「it_infra」に変更</p> <p>デフォルト値の変更後挿入するデータ</p> <table border="1"> <thead> <tr> <th>ID</th> <th>NAME</th> <th>JOB_ID</th> <th>OM_LEVEL</th> </tr> </thead> <tbody> <tr><td>4</td><td>bob</td><td>デフォルト値</td><td>bronze</td></tr> <tr><td>5</td><td>andy</td><td>sa_rep</td><td>silver</td></tr> </tbody> </table> <p>「解答」本文10～11行目</p> <pre>INSERT INTO om_master VALUES(2, 'mary', DEFAULT, 'silver');</pre>	ID	NAME	JOB_ID	OM_LEVEL	1	ben	it_prog	silver	2	mary	デフォルト値	silver	3	tom	sa_rep	gold	ID	NAME	JOB_ID	OM_LEVEL	4	bob	デフォルト値	bronze	5	andy	sa_rep	silver
ID	NAME	JOB_ID	OM_LEVEL																																																								
1	ben	it_prog	silver																																																								
2	mary	default	silver																																																								
3	tom	sa_rep	gold																																																								
ID	NAME	JOB_ID	OM_LEVEL																																																								
4	bob		bronze																																																								
5	andy	sa_rep	silver																																																								
ID	NAME	JOB_ID	OM_LEVEL																																																								
1	ben	it_prog	silver																																																								
2	mary	デフォルト値	silver																																																								
3	tom	sa_rep	gold																																																								
ID	NAME	JOB_ID	OM_LEVEL																																																								
4	bob	デフォルト値	bronze																																																								
5	andy	sa_rep	silver																																																								
74	283ページ 下から4行目	一緒に設定されいなければ、	一緒に設定されていなければ、																																																								
75	286ページ 図10-9内下端	INSERT INTO employees VALUES(203, 'Scott', 2); 行1でエラーが発生しました。: ORA-02292: 整合性制約(HR.EMP_DEPT_FK)に違反しました - 親キーがあり	INSERT INTO employees VALUES(203, 'Scott', 2); 行1でエラーが発生しました。: ORA-02291: 整合性制約(HR.EMP_DEPT_FK)に違反しました - 親キーがあり																																																								
76	288ページ 下から6行目	CONSTRAINT emp_fk FOREIGN KEY department_id	CONSTRAINT emp_fk FOREIGN KEY (department_id)																																																								
77	289ページ 表10-6の「説明」欄最下行	NULLする	NULLにする																																																								
78	290ページ下から17行目 291ページ8行目	(P290の例の見出し) CHECK制約の設定例1 (P291の例の見出し) PRIMARY KEY制約の設定例2	(P290の例の見出し) CHECK制約の設定例 (P291の例の見出し) 表作成後に制約を設定する例																																																								
79	290ページ 最下行	email VARCHAR2(30),	email VARCHAR2(30), salary NUMBER,																																																								
80	310ページ 12行目 （「順序のキャッシュ」最下	メモリー内の順序値が消失します。	メモリー内の順序値が消失します。																																																								
81	311ページ9行目 314ページ15行目 （311ページ「CREATE SEQUENCE 文の構文」リスト5行目） （314ページ「ALTER SEQUENCE 分の構文」リスト4行目）	[MINVALUE 整数値 NOMAXVALUE]	[MINVALUE 整数値 NOMINVALUE]																																																								
82	311ページ 下から4行目	達しとき、	達したとき、																																																								

連番	ページと該当箇所	誤 赤字：修正箇所（修正前） 緑字：説明その他	正 青字：修正箇所（修正後） 緑字：説明その他
83	312ページ 下から17行目 （「順序の使用例」リスト2行目）	メモリーにキャッシュは10とする。	メモリーへのキャッシュは10とする。
84	314ページ 最下行	以下に順序を変更例をあげます。	以下に順序の変更例をあげます。
85	317ページ 下から14行目 （「ここに着眼」本文8行目）	03-05-15',	03-MAY-15',
86	318ページ 8～21行目 （「解答」7行目以後）	1 INSERT INTO managers VALUES (mgr_seq.nextval, 'Alfred', 'Shaffer', '96-03-11', 12000, 'SA_MAN', 80); 2 INSERT INTO managers VALUES (mgr_seq.nextval, 'David', 'Barnes', '00-10-15', 10000, 'MK_MAN', 20); 3 INSERT INTO managers VALUES (mgr_seq.nextval, 'Frank', 'Reed', '05-08-01', 10000, 'IT_PROG', 60); 4 INSERT INTO managers VALUES (mgr_seq.nextval, 'Rex', 'Erickson', '02-07-30', 12000, 'PU_MAN', 30); 5 INSERT INTO managers VALUES (mgr_seq.nextval, 'James', 'Mitchell', '99-12-01', 8000, 'ST_MAN', 100); ALTER SEQUENCE mgr_seq MAXVALUE 220; 6 INSERT INTO managers VALUES (mgr_seq.nextval, 'Charles', 'Swearingen', '03-05-15', 11200, 'SA_MAN', 80);	1 INSERT INTO managers VALUES (mgr_seq.nextval, 'Alfred', 'Shaffer', '96-MAR-11', 12000, 'SA_MAN', 80); 2 INSERT INTO managers VALUES (mgr_seq.nextval, 'David', 'Barnes', '00-OCT-15', 10000, 'MK_MAN', 20); 3 INSERT INTO managers VALUES (mgr_seq.nextval, 'Frank', 'Reed', '05-AUG-01', 10000, 'IT_PROG', 60); 4 INSERT INTO managers VALUES (mgr_seq.nextval, 'Rex', 'Erickson', '02-JUL-30', 12000, 'PU_MAN', 30); 5 INSERT INTO managers VALUES (mgr_seq.nextval, 'James', 'Mitchell', '99-DEC-01', 8000, 'ST_MAN', 100); ALTER SEQUENCE mgr_seq MAXVALUE 220; 6 INSERT INTO managers VALUES (mgr_seq.nextval, 'Charles', 'Swearingen', '03-MAY-15', 11200, 'SA_MAN', 80);
87	318ページ 下から6～3行目	索引がない表では、データの一意性とソートの順番が保証されていないため、最後の行まで全件走査する必要があります。目的のデータがたまたま表の先頭にあった場合でも、データの一意性の保証されないため、全件走査	索引がない表では、データの一意性とソートの順番が保証されていないため、最後の行まで全表走査します。目的のデータがたまたま表の先頭にあった場合でも、データの一意性が保証されないため、全表走査します。
88	320ページ 11行目 （索引の削除例）	CREATE INDEX empid_cp_idx ON emp_copy (employee_id);	DROP INDEX empid_cp_idx;
89	321ページ 2～3行目	・表の規模が大きく、ほとんどの問合せで表内の2～4%未満の行しか戻らないと予想できる場合	（2行分削除）