

예상문제은행 정답 및 해설

2과목 스프레드시트 일반

3과목 데이터베이스 일반





1장 정답 및 해설 — 입력 및 편집

1.② 2.③ 3.② 4.④ 5.② 6.① 7.③ 8.④ 9.② 10.③ 11.② 12.③ 13.② 14.②
 15.③ 16.④ 17.④ 18.② 19.③ 20.③ 21.② 22.④ 23.③ 24.① 25.③ 26.③ 27.② 28.③
 29.② 30.④ 31.① 32.② 33.④ 34.② 35.② 36.③ 37.① 38.② 39.③ 40.② 41.④ 42.③
 43.③ 44.③ 45.① 46.③

1. Section 066

현재 셀 포인터의 위치가 [A1] 셀이므로 [Ctrl]+[;]를 누르고 [Enter]를 누르면 [A1] 셀에 오늘의 날짜가 입력되고, [Ctrl]+[;]에 이어 [Ctrl]+[Enter]를 누르면 [A1:B3] 영역에 모두 오늘의 날짜가 입력된다.

※ 범위를 지정했을 때는 다른 부분과 달리 하얀색으로 표시되는 부분이 현재 셀 포인터의 위치이다.

2. Section 066

여러 셀에 동일한 내용을 입력하려면 해당 셀을 범위로 지정 후 데이터를 입력하고 [Ctrl]+[Enter]를 눌러야 한다.

3. Section 065

- ① 워크시트를 복사하려면 [Ctrl]을 누른 채 원본 워크시트 탭을 마우스로 드래그 앤 드롭하면 된다.
- ③ 연속된 여러 개의 시트를 선택할 때는 첫 번째 시트를 선택하고 [Shift]를 누른 상태에서 마지막 워크시트의 시트 탭을 클릭하면 된다.
- ④ 떨어져 있는 여러 개의 시트를 선택할 때는 먼저 [Ctrl]을 누른 상태에서 원하는 워크시트의 시트 탭을 차례로 클릭하면 된다.

4. Section 066

- 통화 기호는 수치 데이터의 일부로 인식된다.
- ②의 %73, ④의 \$625와 같은 수치 데이터를 문자열로 인식시키려면, ②와 같이 데이터 앞에 작은따옴표(')를 입력해야 한다.

5. Section 072

- ① @@@ : @은 문자 데이터의 표시 위치를 지정하는 것으로, @를 표시할 때마다 해당 문자 데이터가 표시되므로 '컴칼컴칼컴칼'로 표시된다.

- ② #“0,000” : #은 유효한 숫자 1자리를 표시하므로, 숫자 한 자리를 표시한 후 그 뒤에 “0,000” 텍스트를 표시하면 10,000로 표시된다.

- ③ [<=1]0.0%;G/표준 : 셀에 입력된 값이 1보다 작거나 같으면 0.0%로 표시하고, 그렇지 않으면 G/표준 형식으로 표시한다. 0.5는 1보다 작으므로 0.0% 형식이 적용되어 50.0%로 표시된다.

- 0.0% : 숫자에 100을 곱한 다음 소수점 첫째자리까지 표시한 후 %를 붙임

- ④ mmm-dd : 월은 Jan~Dec와 같이 영문 세 자리로 표시하고 일은 두 자리로 표시하므로, Mar-01로 표시된다.

6. Section 065

워크시트를 복사하려면 [Alt]가 아니라 [Ctrl]을 누르면서 원본 워크시트 탭을 마우스로 이동시켜야 한다.

7. Section 066

시간은 기본적으로 24시간제로 표시되며, 12시간제로 표시할 때는 시간 뒤에 한 칸 띄우고 'AM'이나 'PM'을 입력한다.

8. Section 066

셀에 12-12-12를 입력하면 날짜로 인식되어 2012-12-12로 표시된다.

9. Section 066

A열의 채우기 간격은 1이므로 [A4], [A5], [A6] 셀에 '목', '금', '토'가 입력되고, B열의 채우기 간격은 2이므로 [B4], [B5], [B6] 셀에는 '일', '화', '목'이 입력된다.

10. Section 069

마우스 포인터를 열 머리글의 왼쪽 경계선에 위치시킨 후 더블 클릭하면 경계선 왼쪽 열의 열 너비가 해당 열에 입력된 데이터

24. Section 066

- [A2:C4] 영역을 범위로 지정한 후 조건부 서식을 지정하였으므로 [A2:C4] 영역에서 조건에 맞는 셀에 서식을 지정한다.
 - 조건에 입력한 수식의 열 이름 앞에 '\$'가 붙어 있으므로 조건에 맞는 데이터가 있는 행 전체에 서식이 지정된다.
 - 조건 1 : =MAX(\$B\$2:\$B\$4)=\$B2 → [B2:B4] 영역에서 가장 큰 값은 [B3] 셀이다. [A3:C3] 영역에 조건부 서식이 적용된다.
 - 조건 2 : =LEFT(\$A2,1)="손" → 성명의 왼쪽 한 글자가 "손"인 셀은 [A3] 셀이다. [A3:C3] 영역에 조건부 서식이 적용된다.
- ∴ 여러 개의 규칙을 모두 만족할 경우 규칙에 지정된 서식이 모두 적용된다. 단 서식이 충돌할 경우에는 우선 순위가 높은 규칙에 지정된 서식이 적용된다.

25. Section 069

너비를 변경하고자 하는 셀이 속한 열과 오른쪽 열과의 머릿글 경계선을 더블클릭하면 열에 입력된 가장 긴 문자열의 길이에 맞게 열 너비가 조절된다.

26. Section 070

- ① [테두리 없음]을 이용하면 복사한 셀에 적용된 테두리만 제외하고 모두 붙여넣는다.
- ② [연결하여 붙여넣기]를 이용하면 복사한 원본 셀과 붙여넣기한 셀의 내용을 서로 연결하여 붙여넣는다. 단 서식은 복사되지 않는다.
- ④ [그림 형식] → [그림으로 붙여넣기]를 이용하면 복사한 셀에 입력된 내용이 변경되어도 그림에 표시된 텍스트는 변경되지 않는다. 복사한 셀에 입력된 내용이 변경되면 그림에 표시된 텍스트도 자동으로 변경되도록 하려면 [그림 형식] → [연결하여 그림 붙여넣기]를 이용해야 한다.

27. Section 070

- ① 연속된 여러 개의 시트를 선택할 때는 첫 번째 시트를 선택하고 [Shift]를 누른 상태에서 원하는 마지막 워크시트의 시트 탭을 클릭하면 된다.
- ③ 다음 워크시트로 전환하려면 [Ctrl] + [PgDn]을 누르고, 이전 워크시트로 전환하려면 [Ctrl] + [PgUp]을 누르면 된다.
- ④ 떨어져 있는 여러 개의 시트를 선택할 때는 [Ctrl]을 누른 채로 원하는 워크시트의 시트 탭을 클릭한다.

28. Section 070

이전 시트로 이동하려면 [Ctrl] + [PgUp], 다음 시트로 이동하려면 [Ctrl] + [PgDn]을 누르면 된다.

※ 시트 탭의 색 변경 : 시트 탭의 바로 가기 메뉴 중 [탭 색]에서 변경할 색 선택

29. Section 070

공유 통합 문서에서 변경 내용이 충돌할 경우 적용할 내용을 확인한 후 선택하여 저장하거나 변경 내용이 무조건 저장되도록 지정할 수 있다.

30. Section 070

통합 문서의 일부 시트만 선택하여 웹 페이지로 저장할 수 있다.

31. Section 065

[Shift] + [F10]을 누르면 워크시트의 셀 포인터 위치에 바로 가기 메뉴가 표시된다. 리본 메뉴에 바로 가기 키가 표시되도록 하려면 [Alt] 또는 [F10]을 눌러야 한다.

32. Section 070

- ① 일반적인 서식 파일의 확장자는 'xltx'이고, 매크로가 포함된 서식 파일의 확장자는 'xltxm'이다.
- ③ 기본 서식 파일을 새로 만들려면 워크시트는 'Sheet', 통합 문서는 'Book'으로 파일명을 지정한다.
- ④ 행, 열, 시트 등을 숨기거나 워크시트, 셀을 변경하지 못하게 통합 문서를 보호하여 서식 파일을 작성할 수 있다.

33. Section 070

통합 문서 공유가 설정된 파일을 다른 위치에 복사해도 공유 설정값은 유지된다.

34. Section 070

보호 대상으로 구조와 창이 있는 것은 [검토] → [변경 내용] → [통합 문서 보호] → [구조 및 창 보호]이다.

35. Section 072

사용자 지정 서식 코드 중 '*'는 '*' 기호 다음에 있는 특정 문자를 셀의 너비만큼 반복하여 채워 표시하라는 기호이다. 즉 123이 입력된 셀에 *0##,##0으로 표시형식을 지정하면 '*' 기호 다음에 있는 '0'을 셀의 너비만큼 반복하여 표시하므로 00000123으로 표시된다.

※ '*'와 '0'의 글자 크기를 비교하였을 경우 '0'이 '*'보다 크기 때문에 사용자 지정 표시 형식을 '*##,##0'로 지정하면 '*****123'과 같이 '*'가 7개 표시되지만 '*0##,##0'으로 지정하면 '00000123'과 같이 '0'이 5개만 표시된다.

37. Section 072

- ① 15:05:20 → h:m:s AM/PM → 시간(h), 분(m), 초(s)를 한 글자로 표시하되, 12시각제로 표시하므로 '3:5:20 PM' 으로 표시된다.
- ② 20:30 → hh:mm AM/PM → 시간(hh)과 분(mm)을 두 글자로 표시하되, 12시각제로 표시하므로 '08:30 PM' 으로 표시된다.
- ③ 0.5 → hh:mm → 0.5는 하루의 반을 의미하므로 12:00으로 표시된다.
- ④ 35:30 → [h]:mm → [h]는 현재 시간이 아닌 경과된 시간을 표시하므로 35:30 그대로 표시된다.

38. Section 072

- ②의 결과는 05-Jan이다.

39. Section 071

셀에 맞춤

텍스트의 글꼴 크기를 줄여 한 셀에 텍스트를 모두 표시한다.

40. Section 071

문자열의 길이가 셀의 열 너비에 맞도록 설정하는 기능은 있지만 문자열의 길이에 맞춰 셀의 열 너비가 자동으로 조절되게 설정하는 기능은 따로 없다. 문자열의 길이가 셀의 열 너비에 맞도록 설정하려면 '셀 서식' 대화상자의 '맞춤' 탭에서 '셀에 맞춤'을 선택한다.

41. Section 067

셀 포인터가 포함된 데이터 목록 전체를 선택할 때는 **Ctrl** + ***** 나 **Ctrl** + **Shift** + **Spacebar** 를 누르면 된다.

42. Section 073

[A1:C8] 영역을 범위로 지정한 상태에서 조건부 서식을 실행하여 규칙을 '중복'으로 지정하면 [A1:C8] 영역에서 동일한 데이터가 두 번 이상 들어간 셀에 지정한 서식이 적용된다. [A1:C8] 영역에서 중복으로 입력된 데이터는 "합격"과 "불합격"이므로, "합격"과 "불합격"이 입력된 14개 셀에 모두 지정된 서식이 적용된다.

43. ① 표시할 텍스트 : 하이퍼링크를 설정한 셀에 항상 표시되는 문자열

- ② 스크린 팁 : 하이퍼링크 위에 마우스 포인터를 놓았을 때 표시되는 문자열
- ④ 연결 대상 : 삽입할 링크 종류 선택, 기존 파일이나 웹 페이지에 대한 링크를 만들려면 '기존 파일/웹 페이지'를 선택해야 함

44. Section 070

- 통합 문서를 공유하려면 [검토] → [변경 내용] → [통합 문서 공유]를 클릭하고 '편집' 탭에서 '여러 사용자가 동시에 변경할 수 있으며 통합 문서 병합도 가능' 확인란을 선택해야 한다.
- '변경 내용 추적'은 공유를 설정한 후 변경된 내용의 보관 여부를 지정하는 옵션이다.

45. Section 072

문제에 제시된 내용을 차례대로 표현하면 다음과 같다.

- 입력한 문자 앞에 "판매액"을 표시 : "판매액"@
 - 0이면 0.00으로 표시 : 0.00
 - 양수이면 검은색으로 천 단위 구분 기호를 사용해서 표시하고, 소수 자릿수는 둘째 자리까지 표시 : [검정]#,##0.00
 - 음수이면 빨간색으로 천 단위 구분 기호와 괄호를 표시하며, 소수 자릿수는 둘째 자리까지 표시 : [빨강](#,##0.00)
- 위의 서식을 사용자 지정 서식의 순서에 맞게 양수, 음수, 0, 텍스트 순으로 표시하면 다음과 같다.

[검정]#,##0.00;[빨강](#,##0.00);0.00;"판매액"@

46. Section 072

문제의 서식 지정 조건에 따라 각 조건을 하나씩 구분하여 지정해 보면 다음과 같다.

구분	기준	표시문자열	문자색	입력값 표시
조건1	500 이상	목표달성!!	파랑색	510
서식	[>=500]	"목표달성!!"	[파랑]	G/표준
종합	[파랑]>=500;"목표달성!!"G/표준			
조건2	300 미만	주의요망!!	빨강색	250
서식	[<300]	"주의요망!!"	[빨강]	G/표준
종합	[빨강]<300;"주의요망!!"G/표준			
조건3	300-499	노력요망!!	표준 색	400
서식		"노력요망!!"	[검정]	G/표준
종합	[검정]300<=500;"노력요망!!"G/표준			

※ 각 조건에 대한 서식을 하나로 합치면 아래와 같다.

[파랑]>=500;"목표달성!!"G/표준;[빨강]<300;"주의요망!!"G/표준;[검정]300<=500;"노력요망!!"G/표준

2장 정답 및 해설 — 수식 활용

1.③ 2.① 3.② 4.③ 5.① 6.④ 7.③ 8.④ 9.④ 10.① 11.① 12.③ 13.① 14.①
15.② 16.④ 17.② 18.① 19.④ 20.② 21.③ 22.④ 23.① 24.④ 25.② 26.③ 27.③ 28.④
29.② 30.③ 31.② 32.① 33.④ 34.① 35.② 36.① 37.② 38.④ 39.④ 40.② 41.②

1. Section 074

수식의 결과값은 해당 셀에 표시되고, 수식은 수식 입력줄에 표시된다.

2. Section 075

다른 통합 문서를 참조할 경우 파일 이름은 대괄호([])로 묶고, 시트명과 셀 주소는 느낌표(!)로 구분한다.

3. Section 075

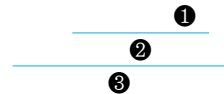
여러 셀에 공통적으로 적용될 고정된 셀은 절대 참조로, 변경되어 적용될 셀은 상대 참조로 입력해야 한다. 즉 합계가 입력된 [E2] 셀은 절대 참조(\$E\$2)를, 수업 시간(B2, C2, D2)은 상대 참조(B2)를 적용해야 한다.

4. Section 077

① =FIXED(40899.4928, -2, TRUE) : FIXED(인수, 자릿수, 논리값)은 인수를 반올림하여 지정된 자릿수까지 텍스트로 표시하는 함수이다. 논리값이 TRUE이면 십표를 포함하지 않으므로 40899.4928를 백의 자리(-2)로 반올림하면 40900이 표시된다.

② =EOMONTH(DATE(1900,1,1), 1) : EOMONTH(날짜, 월수)는 지정한 날짜를 기준으로 몇 개월 이전 또는 이후 달의 마지막 날짜의 일련번호를 구하는 함수이다. 1900년 1월 1일을 기준으로 1개월 이후 달의 마지막 날짜는 1900년 2월 28일이다. 1900년 1월 1일을 기준으로 1900년 2월 28일의 일련번호를 구하면 1월의 31일에 2월의 28일을 더한 59가 된다.

③ =FACT(TRUNC(PI()))



① PI() : 수치 상수 파이(π)를 15자리(3.14159265358979)까지 나타낸다.

② TRUNC(①) → TRUNC(3.14159265358979) : TRUNC(인수, 자릿수)은 지정된 자릿수 이하를 잘라내는 함수인데, 자릿수가 지정되지 않았으므로 소수점 이하를 모두 버리고 정수 3을 반환한다.

③ FACT(②) → FACT(3) : FACT(인수)는 인수의 계승값을 구하는 함수로, 3×2×1은 6이다.

④ =ROUNDUP(PI(),0) : ROUNDUP(인수, 올림 자릿수)은 지정된 자릿수로 올림하는 함수로, 파이(π) 3.14159265358979를 일(0)의 자리로 올림하면 4가 된다.

5. Section 081

1, 2, 3과 3, 2, 1을 곱한 결과 3, 4, 3의 합계가 계산되어 10이 산출된다.

$$\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \times \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix} = \begin{pmatrix} 3 \\ 4 \\ 3 \end{pmatrix} \rightarrow =\text{SUM}(3,4,3) \rightarrow 10$$

6. Section 078

- 둘 중 하나만 조건을 만족하면 되므로 OR 함수로 두 조건을 묶는다.
- 텍스트 함수 MID()를 이용해 추출한 주민등록번호 8번째 자리("1", "2", "3", "4")는 문자이므로 큰따옴표(" ")로 묶어서 비교해야 한다.
- =IF(OR(MID(A2,8,1)="1", MID(A2,8,1)="3"), "남자", "여자")
 - ① MID(A2,8,1) : 주민등록번호의 8번째 위치에서부터 1문자를 추출한다.
 - ② OR(①="1", ①="3") : ①에서 추출한 문자가 1이거나 3이면 참, 그렇지 않으면 거짓
 - ③ IF(②, "남자", "여자") : ②의 결과값이 참이면 "남자"를, 거짓이면 "여자"를 표시

7. Section 077

① =ABS(INT(-7.9))



① INT(-7.9) : -7.9보다 크지 않은 정수를 구한다 (-8).

② ABS(①) : -8을 절대값으로 표시한다(8).

$$\textcircled{2} = \text{SUM}(\text{TRUNC}(45,6), \text{MOD}(32, 3))$$



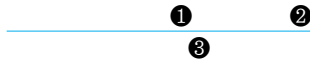
- ① TRUNC(45,6) : 45.6을 소수점 이하를 버리고 정수로 변환한다(45).
 ② MOD(32, 3) : 32를 3으로 나눈 나머지를 구한다(2).
 ③ SUM(①, ②) → SUM(45, 2) : 두 수의 합을 구한다(47).

$$\textcircled{3} = \text{POWER}(\text{ROUND}(2,3, 0), \text{SQRT}(4))$$



- ① ROUND(2,3, 0) : 2.3을 반올림하여 정수로 표시한다(2).
 ② SQRT(4) : 4의 양의 제곱근을 구한다(2).
 ③ POWER(①, ②) → POWER(2, 2) : 2², 즉 2×2의 4를 반환한다.

$$\textcircled{4} = \text{CHOOSE}(3, \text{SUM}(10, 10), \text{INT}(30.50), 50)$$



- ① SUM(10, 10) : 두 수의 합을 구한다(20).
 ② INT(30.50) : 30.50보다 크지 않은 정수를 구한다(30).
 ③ CHOOSE(3, ①, ②, 50) → CHOOSE(3, 20, 30, 50) : 20, 30, 50 중 세 번째에 있는 값을 입력한다(50).

8. Section 074

수식에서 직접 또는 간접적으로 수식이 입력된 그 셀을 그 수식에서 참조하는 경우를 순환 참조라고 한다. 수식에서 수식을 입력한 [C1] 셀을 더하기 때문에 아래와 같은 순환 참조 경고 메시지가 표시된다.



9. Section 077

- SUMPRODUCT(배열1, 배열2) : 배열에서 대응하는 요소를 모두 곱하고 그 곱의 합을 구한다.

$$\therefore =\text{SUMPRODUCT}(\{1,2,3\}, \{5,6,7\})$$

$$= \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \times \begin{pmatrix} 5 \\ 6 \\ 7 \end{pmatrix} = 1 \times 5 + 2 \times 6 + 3 \times 7 = 38$$

10. Section 074

②번은 #DIV/0!, ③번은 #N/A, ④번은 #REF! 오류에 대한 설명이다.

11. Section 081

배열 상수로 %는 사용할 수 없다. 하지만 ‘=10%*100’과 같이 %를 수식에 사용하면 이 수식 자체가 배열 상수로 들어가는

것이 아니라 수식의 결과값인 ‘10’이 상수로 들어가기 때문에 오류가 아닌 ‘10’이 표시된다.

12. Section 077

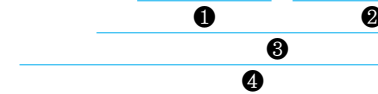
① =AVERAGE(A1:A6) : AVERAGE(인수1, 인수2, ...)는 인수의 평균을 구하는 함수로, 결과는 1.25(5/4)가 된다.

$$\textcircled{2} = \text{ROUNDUP}(\text{AVERAGE}(\text{SCORES}), 0)$$



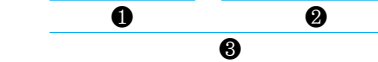
- ① AVERAGE(SCORES) : SCORES로 이름 정의된 영역(A1:A6)의 평균을 구한다(1.25).
 ② ROUNDUP(①, 0) : ①을 소수 이하 첫째 자리에서 올림하면 결과는 2가 된다.

$$\textcircled{3} = \text{ROUND}(\text{INT}(\text{SUM}(\text{SCORES}) / \text{COUNT}(\text{SCORES})), 0)$$



- ① SUM(SCORES) : SCORES로 이름 정의된 영역(A1:A6)의 합계를 구한다(5).
 ② COUNT(SCORES) : SCORES로 이름 정의된 영역(A1:A6)에서 숫자가 들어 있는 셀의 개수를 센다(4).
 ③ INT(① / ②) : ①을 ②로 나눈 후 정수만 표시한다(5/4는 1.25, 1.25에서 정수만 표시하면 결과는 1).
 ④ ROUND(③, 0) : ③을 소수 이하 첫째 자리에서 반올림한다. 결과는 1이 된다.

$$\textcircled{4} = \text{SUM}(\text{SCORES}) / \text{COUNTA}(\text{SCORES})$$



- ① SUM(SCORES) : SCORES로 이름 정의된 영역(A1:A6)의 합계를 구한다(5).
 ② COUNTA(SCORES) : SCORES로 이름 정의된 영역(A1:A6)에서 데이터가 들어 있는 셀의 개수를 센다(5).
 ③ ① / ② : ①을 ②로 나누면 결과는 1이 된다.

13. Section 081

조건이 2개일 때 배열 수식을 이용하여 개수를 구하는 방법은 다음의 3가지 방법이 있다.

- 방법1 : {=SUM((조건1)*(조건2))}
- 방법2 : {=SUM(IF(조건1, IF(조건2,1)))}
- 방법3 : {=COUNT(IF((조건1)*(조건2),1))}

1. 조건과 범위 찾기

- 조건1 : 직책이 “부장”이란 조건은, 비교 대상이 될 직책이 있는 범위와 비교할 기준이 되는 “부장”을 “=”으로 연결하여 적어주면 된다(B2:B8=“부장”).

- 조건2 : 기본급이 1,000,000 이상이란 조건은, 비교 대상이 될 기본급이 있는 범위와 비교할 기준이 되는 1,000,000을 ">="로 연결하여 적어주면 된다(C2:C8)=1000000).

2. 위의 조건을 개수 구하기 배열 수식에 대입하면 다음과 같다.

- 방법1 : =SUM((B2:B8="부장")*(C2:C8)=1000000))
- 방법2 : =SUM(IF(B2:B8="부장", IF(C2:C8= 1000000, 1)))
- 방법3 : =COUNT(IF((B2:B8="부장")*(C2:C8)>=1000000),1)

COUNT는 인수 중 숫자가 들어 있는 셀의 개수를 구하는 함수로 방법3을 '=COUNT(IF((B2:B8="부장")*(C2:C8)>=1000000),1, "")' 으로 입력해도 결과값은 동일하다.

※ 이 문제는 하나의 값을 구하는 수식으로 범위를 절대 참조로 지정하거나 지정하지 않아도 결과값은 동일하다. 수식을 입력한 후 **[Ctrl]+[Shift]+[Enter]**를 누르면 중괄호 { }가 표시되어 '{=COUNT(IF((B2:B8="부장")*(C2:C8)>=1000000),1, ""))}' 혹은 '{=COUNT (IF(\$B\$2:\$B\$8="부장")*(\$C\$2:\$C\$8 >= 1000000), 1, ""))}' 로 표시된다.

궁금해요!



시나공 Q&A 베스트

Q {=COUNT(IF(\$B\$2:\$B\$8="부장")*(\$C\$2:\$C\$8)=1000000),1, "")}에서 마지막에 내용없이 큰따옴표 두 개만 입력했는데 왜 그런 건가요?

A IF 함수에서 조건을 만족하면 1을, 그렇지 않으면 널("")을 넣은 후 COUNT 함수로 1의 개수를 구해 조건에 만족하는 전체 개수를 구하는 수식입니다. COUNT 함수는 숫자의 개수를 세는 함수이므로 널("")을 넣든, 안 넣든 결과는 동일합니다.

14. Section 078

=LEFT(A2, LEN(A2)-1)

①
②

- ① LEN(A2) : [A2] 셀의 문자 개수를 구함(4)
- ② LEFT(A2, ①-1) → LEFT(A2, 4-1) : [A2] 셀에 입력된 텍스트 중 왼쪽부터 3글자를 표시함("서귀포")

15. Section 080

'=IF(ISBLANK(B1), "", A1*C1)' 은 [B1] 셀이 비어 있으면 "" (빈 값)을 출력하고, 아니면 [A1] 셀과 [C1] 셀의 값을 곱한 결과를 출력한다.

16. Section 077

① =SUM(LARGE(①,3,7,8),2),POWER(2,4),MOD(25,3))

① ② ③
④

① LARGE(①,3,7,8),2) : 9, 3, 7, 8 중 두 번째로 큰 값을 반환한다(8).

② POWER(2,4) : 2⁴, 즉 2×2×2×2의 값 16을 반환한다.

③ MOD(25,3) : 25를 3으로 나눈 나머지를 반환한다(1).

④ SUM(①,②,③) → SUM(8,16,1) : 8, 16, 1을 모두 더한 값을 반환한다(25).

② =AVERAGE(LARGE(①,3,7,8),3),5)

①
②

① LARGE(①,3,7,8),3) : 9, 3, 7, 8 중 세 번째로 큰 값을 반환한다(7).

② AVERAGE(①,5) → AVERAGE(7,5) : 7과 5의 평균을 반환한다(6).

③ =MAX(MOD(ABS(-5),3),ABS(MOD(-5,3)))

① ③
② ④
⑤

① ABS(-5) : -5의 절대값을 표시한다(5).

② MOD(①,3) → MOD(5,3) : 5를 3으로 나눈 나머지를 반환한다(2).

③ MOD(-5,3) : -5를 3으로 나눈 나머지를 반환한다(1).

④ ABS(③) → ABS(1) : 1의 절대값을 표시한다(1).

⑤ MAX(②,④) → MAX(2,1) : 2와 1 중 최대값을 반환한다(2).

④ =COUNT(1, "가", SMALL(①,2,3,4),2))

①
②

① SMALL(①,2,3,4),2) : 12, 3, 4 중 두 번째로 작은 값을 반환한다(4).

② COUNT(1, "가", ②) → COUNT(1, "가", 4) : 1, "가", 4 중 수치 데이터의 개수를 반환한다(2).

17. Section 079

=AVERAGE(INDEX(점수, 2, 1),MAX(점수))

① ②
③

① INDEX(점수, 2, 1) → INDEX(B2:D6, 2, 1) : [B2:D6] 영역에서 2행 1열에 있는 데이터를 표시한다(75).

② MAX(점수) → MAX(B2:D6) : [B2:D6] 영역에서 가장 큰 값을 표시한다(100).

③ AVERAGE(①, ②) → AVERAGE(75, 100) : 두 수의 평균을 구한다(87.5).

18. Section 077

MMULT는 행렬의 곱을 구하는 함수이고, 행렬의 합을 구하는 함수는 없다.

19. Section 077

- ① =COUNTA((고객구분="정회원")*(지역="인천")) : COUNTA(인수1, 인수2, ...) 함수는 인수 중 비어 있지 않은 셀의 개수를 구하는 함수로 '(고객구분="정회원")*(지역="인천")'를 하나의 데이터로 인식하여 결과값은 1이 된다.
- ② =COUNTIF((고객구분="정회원")*(지역="인천")) : COUNTIF(범위, 조건) 함수는 지정된 범위에서 조건에 맞는 셀의 개수를 구하는 함수로 '(고객구분="정회원")*(지역="인천")'를 첫 번째 인수 범위로 인식하고, 두 번째 조건을 지정하지 않았으므로 인수가 부족하다는 오류 메시지가 표시된다.
- ③ =COUNT((고객구분="정회원")*(지역="인천")) : COUNT(인수1, 인수2, ...) 함수는 인수 중 숫자가 들어 있는 셀의 개수를 구하는 함수로 '(고객구분="정회원")*(지역="인천")'을 하나의 데이터, 즉 문자 데이터로 인식하므로 결과값은 0이 된다.
- ④ =SUMPRODUCT((고객구분="정회원")*(지역="인천")) : SUMPRODUCT(배열1, 배열2, ...)는 배열에서 대응하는 요소를 모두 곱하고 그 곱의 합을 구하는 함수이고, '고객구분'은 [B2:B7], '지역'은 [C2:C7]이므로 다음과 같이 계산된다.

$$\begin{aligned}
 &=SUMPRODUCT \begin{pmatrix} (B2="정회원")*(C2="인천") \\ (B3="정회원")*(C3="인천") \\ (B4="정회원")*(C4="인천") \\ (B5="정회원")*(C5="인천") \\ (B6="정회원")*(C6="인천") \\ (B7="정회원")*(C7="인천") \end{pmatrix} \\
 &\quad \downarrow \\
 &=SUMPRODUCT \begin{pmatrix} (FALSE)*(TRUE) \\ (TRUE)*(FALSE) \\ (TRUE)*(TRUE) \\ (FALSE)*(TRUE) \\ (TRUE)*(TRUE) \\ (FALSE)*(FALSE) \end{pmatrix} \\
 &\quad \downarrow \\
 &=SUMPRODUCT \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} \\
 &=2
 \end{aligned}$$

※ SUMPRODUCT 함수는 각 인수를 콤마(,)로 구분하는데, 이 문제는 콤마로 구분되어 있지 않으므로 '(고객구분="정회원")*(지역="인천")'을 하나의 배열로 인식하므로 곱하는 작업 없이 합계를 구함

20. Section 081

개수를 구하는 배열 수식은 항상 다음의 세 가지 식이 있으며, 조건의 개수에 따라 조건을 지정하는 부분만 늘어난다. 이 문제는 조건이 하나이므로 조건이 하나일 때의 조건 지정 방법은 다음과 같다.

- 방법1 : {=SUM((조건) * 1)}
- 방법2 : {=SUM(IF(조건, 1))}
- 방법3 : {=COUNT(IF(조건, 1))}

1. 조건과 범위 찾기

- 조건1 : '컴활2급을 신청한' 이란 조건은, 비교 대상이 될 시험종류가 있는 범위와 비교할 기준이 되는 "컴활2급"을 "<"으로 연결하여 적어주면 된다(C2:C6="컴활2급").

2. 위의 조건을 개수 구하기 배열 수식에 대입하면 다음과 같다.

- 방법1 : =SUM((C2:C6="컴활2급") * 1)
- 방법2 : =SUM(IF(C2:C6="컴활2급", 1))
- 방법3 : =COUNT(IF(C2:C6="컴활2급", 1))

방법 2에서 SUM은 합계를 구하는 함수이므로 '=SUM(IF(C2:C6="컴활2급",1,0))'으로 입력해도 결과는 동일하다. 수식을 입력한 후 [Ctrl] + [Shift] + [Enter]를 누르면 중괄호 {}가 표시되어 '{=SUM(IF(C2:C6="컴활2급", 1, 0))}'로 표시된다.

※ ①번의 경우 COUNT 함수는 수치 데이터의 개수를 구하는 함수인데, 조건이 거짓일 때 입력한 0도 수치 데이터이므로 조건과는 상관없이 전체 개수를 구한다. ①번을 옮겨 수정하면 {=COUNT(IF(C2:C6="컴활2급", 1))}이다.

21. Section 077, 078

총점이 상위 30%라는 것은 총점 중에서 70% 위치에 있는 점수 이상을 말한다. 문제에 주어진 총점 중에서 70%에 위치하는 값을 '=PERCENTILE(D2:D6,0.7)'을 이용하여 계산하면 184.6이다. 즉 총점이 184.6 이상이면 상위 30%에 해당한다. 0.7 대신 70%를 입력해도 된다.

22. Section 074

주요 기능키의 기능

- [F1] : 도움말
- [F2] : 현재 셀을 편집 상태로 표시
- [F5] : '이동' 대화상자 표시
- [F7] : '맞춤법 검사' 대화상자 표시
- [F10] : 리본 메뉴 선택
- [F11] : 현재 범위의 데이터에 대한 기본 차트를 만들

23. Section 079

① =INDEX(A1:C4,MATCH("배",A1:A4,0),1)

①

②

① MATCH("배",A1:A4,0) : [A1:A4] 영역에서 "배"가 입력된 셀의 상대 위치를 계산한다(4).

② =INDEX(A1:C4,①,1) → =INDEX(A1:C4,4,1) : [A1:C4] 영역에서 4행 1열, 즉 [A4] 셀에 입력된 값을 표시한다(배).

② =INDEX(A1:C4,4,2) : [A1:C4] 영역에서 4행 2열, 즉 [B4] 셀에 입력된 값을 출력한다(40).

③ =INDEX(A1:C4,MATCH(2300,C1:C4,0),2)

①

②

① MATCH(2300,C1:C4,0) : [C1:C4] 영역에서 2300이 입력된 셀의 상대 위치를 계산한다(4).

② =INDEX(A1:C4,①,2) → =INDEX(A1:C4,4,2) : [A1:C4] 영역에서 4행 2열, 즉 [B4] 셀에 입력된 값을 출력한다(40).

④ =INDEX(B3:C4,2,1) : [B3:C4] 영역에서 2행 1열, 즉 [B4] 셀에 입력된 값을 출력한다(40).

24. Section 077, 081

① =IF(COUNT(B3:E3)=0, "", AVERAGE(B3:E3)) : [B3:E3] 영역에 수치 데이터가 입력된 셀의 개수가 0이면 빈칸을 입력하고, 그렇지 않으면 [B3:E3] 영역의 평균을 표시한다(빈칸).

② =IF(COUNTA(B3:E3)=0, "", AVERAGE(B3:E3)) : [B3:E3] 영역에 데이터가 입력된 셀의 개수가 0이면 빈칸을 입력하고, 그렇지 않으면 [B3:E3] 영역의 평균을 표시한다(빈칸).

③ =IF(ISERROR(AVERAGE(B3:E3)), "", AVERAGE(B3:E3))

①

②

① ISERROR(AVERAGE(B3:E3)) : [B3:E3] 영역의 평균이 오류이면 'TRUE', 아니면 'FALSE'를 반환한다.

② ①의 값이 참(TRUE)이면 빈칸, 그렇지 않으면 [B3:E3] 영역의 평균 표시한다(빈칸).

④ =IF(ISBLANK(B3:E3)=FALSE, "", AVERAGE(B3:E3)) : [B3:E3] 영역의 셀이 빈칸이 아니면 "", 그렇지 않으면, 즉 빈칸이면 [B3:E3] 영역의 평균 표시한다(#DIV/0!).

25. Section 078

① =REPLACE("monkey",4,6,"ey") : "monkey"에서 4번째 글자부터 6 글자를 "ey"로 변경한다("money"). 4번째부터 끝까지의 글자수가 3이므로 =REPLACE("monkey",4,3,

"ey")로 작성해도 결과값은 동일하다.

② =ISERROR(4/0) : 4를 0으로 나눈 값이 오류이면 TRUE, 아니면 FALSE를 표시한다(TRUE).

※ 4를 0으로 나누면 #DIV/0! 오류가 표시된다.

③ =FIXED(9876.54321,1) : 9876.54321에 콤마 서식을 적용하고, 소수점 둘째 자리에서 반올림하여 첫째 자리까지만 표시한다(9,876.5).

④ =TEXT("2010-3-21","mmm dd, yy") : "2010-3-21"을 "mmm dd, yy" 형식으로 표시한다(Mar 21, 10).

26. Section 078

①은 "GI-T", ②는 "WOOWOO", ④는 "GILBUT"이 입력된다.

27. Section 079

'=OFFSET(A1,2,3)'은 [A1] 셀에서 2행 아래쪽(3행), 3열 오른쪽(D열)에 위치한 데이터(D3)를 표시하라는 의미이다.

28. Section 075

배열 수식에는 3차원 참조를 사용할 수 없다.

29. Section 081

두 배열의 더하기를 실행하면 두 배열의 같은 위치에 대응하는 요소끼리 더하여 동시에 결과를 계산한다.

	A	B	C	D	E
1	1	2		1	2
2	3	4		3	4
3					
4	2	4			
5	6	8			

[A4] 셀 : [A1]+[D1] = 1 + 1 = 2

[B4] 셀 : [B1]+[E1] = 2 + 2 = 4

[A5] 셀 : [A2]+[D2] = 3 + 3 = 6

[B5] 셀 : [B2]+[E2] = 4 + 4 = 8

30. Section 079

=INDEX(\$E\$2:\$E\$4, MOD(A2-1,3)+1)

①

②

① MOD(A2-1,3) : [A2] 셀에 입력된 값에서 1을 뺀 값을 3으로 나눈 나머지를 구한다(0).

② INDEX(\$E\$2:\$E\$4, ①+1) → INDEX(\$E\$2:\$E\$4, 0+1) : [E2:E4] 영역에서 1행에 있는 데이터를 표시한다("기획팀").

31. Section 077

- MINVERSE : 인수로 주어지는 배열의 역행렬을 구한다.
- ②는 MMULT 함수에 대한 설명이다.

32. Section 079

‘최저 낙찰가’를 구하기 위해서는 최저값을 구하기 위한 데이터베이스 함수 DMIN과 낙찰가를 구하기 위한 열 번호 4가 필요하다. 수식을 작성하면 ‘=DMIN (A2:D8, 4, F2:F3)’ 이 된다.

33. Section 077

- ① =SMALL(A1:G1, {3}) : [A1:G1] 영역에서 세 번째로 작은 값을 구하면 결과값은 30이다.
- ② =AVERAGE(SMALL(A1:G1, {1:2:3:4:5}))

①

②

- ① SMALL(A1:G1, {1:2:3:4:5}) : SMALL 함수에서 몇 번째로 작은 값을 구할지를 지정하는 두 번째 인수를 배열 ({1:2:3:4:5})로 지정하면 배열의 각 인수(1:2:3:4:5) 번째로 작은 값을 모두 구한다. 즉, [A1:G1] 영역에서 첫 번째로 작은 값, 두 번째로 작은 값, 세 번째로 작은 값, 네 번째로 작은 값, 다섯 번째로 작은 값을 구하면 결과값은 {10:20; 30:40:50}이다.
- ② AVERAGE(②) → AVERAGE({10:20;30:40:50}) : 배열의 평균을 구하면 결과값은 30이다.
- ③ =LARGE(A1:G1, {5}) : [A1:G1] 영역에서 다섯 번째로 큰 값을 구하면 결과값은 30이다.
- ④ =SMALL(A1:G1, COLUMN(D1))

①

②

- ① COLUMN(D1) : [D4] 셀의 열 번호를 구하면 결과값은 4이다.
- ② SMALL(A1:G1, ②) → SMALL(A1:G1, 4) : [A1:G1] 영역에서 네 번째로 작은 값을 구하면 결과값은 40이다.

34. Section 080

- 문제를 풀기 위해 필요한 PMT 함수의 인수는 ‘이자, 기간, 현재가치, 미래가치, 납입시점’이다.
- 3천만원은 현재 받은 금액이므로 현재가치에 해당된다.
- 결과값을 양수로 출력하기 위해 현재 받은 금액은 음수로 입력한다.

35. Section 080

- 문제를 풀기 위해 필요한 PV 함수의 인수는 ‘이자, 기간, 금액, 미래가치, 납입시점’이다.

- 연이율이므로 12로 나누어서 월이율로 만든다.
- 기간이 20년이므로 12를 곱해 월 단위로 만든다.
- 결과값을 양수로 출력하기 위해 받은 금액은 음수로 입력한다.
- 지불시점이 매달 말이므로 마지막 Type 인수에는 0을 입력하거나 생략할 수 있다.

36. Section 080

- 문제를 풀기 위해 필요한 FV 함수의 인수는 ‘이자, 기간, 금액, 현재가치, 납입시점’이다.
- 연이율이므로 12로 나누어서 월이율로 만든다.
- 기간이 3년이므로 12를 곱해 월 단위로 만든다.
- 불입시점이 매달 말일일 경우 마지막 Type 인수는 0 또는 생략할 수 있다.

37. Section 077, 078

②는 123.66이 표시되고, 나머지는 123.65가 표시된다.

38. Section 078, 079

- ② MID(A1,8,1) : 주민등록번호의 8번째 위치에서부터 1개를 추출한다.
- ③ IF(②="1", "남", "여") : ②에서 추출한 문자가 “1”이면 “남”, 아니면 “여”를 표시한다.

39. Section 079

- 워크시트에 입력된 조건을 이용할 때는 데이터베이스 함수를 사용한다고 보아야 한다.
- 조건에 맞는 값의 합계를 구하는 데이터베이스 함수는 DSUM이다.
- 데이터베이스 함수의 인수는 ‘범위, 필드번호, 조건’ 순이다.

40. Section 077

- ① =SUMIF(B2:B7, "<>0") / COUNTIF(B2:B7, "<>0")

①

②

- ① SUMIF(B2:B7, "<>0") : SUMIF(조건이 적용될 범위, 조건, 합계를 구할 범위) 함수는 조건에 맞는 셀들의 합계를 구하는 함수로, ‘합계를 구할 범위’를 생략하면 ‘조건이 적용될 범위’에서 조건에 맞는 셀의 합계를 구한다. [B2:B7] 영역에서 0이 아닌 셀의 합계를 구하면 2183이다.
- ② COUNTIF(B2:B7, "<>0") : COUNTIF(범위, 조건) 함수는 지정된 범위에서 조건에 맞는 셀의 개수를 구하는 함수로, [B2:B7] 영역에서 0이 아닌 개수를 구하면 4이다.
∴ ①/② → 2183 / 4 = 545.75

② =SUMIF(B2:B7, "<>0") / COUNT(B2:B7)

①

②

① SUMIF(B2:B7, "<>0") : ①번과 동일하므로 2183이다.

② COUNT(B2:B7) : COUNT(인수1, 인수2, ...) 함수는 인수 중 숫자가 들어 있는 셀의 개수를 구하는 함수로, 결과는 6이다.

∴ ①/② → 2183 / 6 = 363.833

③ =AVERAGEIF(B2:B7, "<>0") : AVERAGEIF(조건이 적용될 범위, 조건, 평균을 구할 범위) 함수는 조건에 맞는 셀들의 평균값을 구하는 함수로, '평균을 구할 범위'를 생략하면 '조건이 적용될 범위'에서 조건에 맞는 셀의 평균을 구한다. [B2:B7] 영역에서 0이 아닌 셀들의 평균을 구하면 545.75이다.

④ {=AVERAGE(IF(B2:B7<>0,B2:B7))} : 조건이 하나 일 때의 평균을 구하는 배열 수식으로, [B2:B7] 영역에서 0이 아닌 셀들의 평균을 구하면 545.75이다.

41. Section 077

SUMPRODUCT(배열1, 배열2, ...)는 배열에서 대응하는 요소를 모두 곱하고 그 곱의 합을 구하는 함수로, 각 인수는 콤마(,)로 구분하여 입력한다. 그런데 이 문제는 인수가 콤마로 구분되어 있지 않으므로 '(A1:A100=C1) * (B1:B100=D1)'을 하나의 배열로 인식하여 곱하는 작업 없이 바로 합계를 구한다.

$$=SUMPRODUCT \begin{pmatrix} (A1=C1) * (B1=D1) \\ (A2=C1) * (B2=D1) \\ \dots \\ (A99=C1) * (B99=D1) \\ (A100=C1) * (B100=D1) \end{pmatrix}$$

두 조건이 모두 참이면 True(1), 하나라도 거짓이면 False(0)이며, 이것들의 합계를 구하는 수식이다. 즉 조건을 모두 만족하는 데이터의 개수를 구하는 수식이다.

① =SUMIFS(A1:A100,C1,B1:B100,D1) : SUMIFS(합계를 구할 범위, 조건1이 적용될 범위, 조건1, 조건2가 적용될 범위, 조건2, ...)는 여러 조건에 맞는 셀들의 합계를 구하는 함수로, 인수 지정이 잘못되었다.

② =COUNTIFS(A1:A100,C1,B1:B100,D1) : COUNTIFS(조건1이 적용될 범위, 조건1, 조건2가 적용될 범위, 조건2,...)는 여러 조건에 맞는 셀의 개수를 구하는 함수로, [A1:A100] 영역의 값이 [C1] 셀과 같고 [B1:B100] 영역의 값이 [D1] 셀과 같은 데이터의 개수를 구한다.

③ =AVERAGEIFS(A1:A100,C1,B1:B100,D1) : AVERAGEIFS(평균을 구할 범위, 조건1이 적용될 범위, 조건1, 조건2가 적용될 범위, 조건2, ...)는 여러 조건에 맞는 셀들의 평균값을 구하는 함수로, 인수 지정이 잘못되었다.

④ =SUBTOTAL(SUM,A1:A100,B1:B100) : =SUBTOTAL(사용할 함수 번호, 인수1, 인수2, ...)은 지정한 함수로 인수들의 부분합을 계산하는 함수로, [A1:A100]과 [B1:B100] 영역의 합계를 계산하려면 '=SUBTOTAL(9,A1:A100,B1:B100)'과 같이 'SUM' 대신 함수 번호 9를 지정해야 한다.

3장 정답 및 해설 — 차트 작성

1.④ 2.① 3.① 4.④ 5.③ 6.④ 7.① 8.③ 9.② 10.③ 11.③ 12.④ 13.② 14.③
15.② 16.③ 17.④ 18.③ 19.② 20.③ 21.③ 22.① 23.② 24.④ 25.④ 26.①

1. Section 082

문자 데이터는 차트로 표현할 수 없다. 차트는 수치 데이터를 막대나 선, 도형, 그림 등을 사용하여 시각적으로 표현한 것으로, ①, ②, ③번과 같은 목적으로 사용된다.

2. Section 082

워크시트에서 차트에 사용될 데이터의 범위를 지정한 후 [Alt] + [F1]을 누르면 원본 데이터가 있는 워크시트에 기본 차트가 작성되고, [F11]을 누르면 별도의 차트 시트에 기본 차트가 작성된다.

3. Section 083

이미 작성된 차트의 범례 위치를 변경하려면 범례의 바로 가기 메뉴에서 [범례 서식]을 선택한 후 '범례 서식' 대화상자의 '범례 옵션' 탭에서 범례의 위치를 변경하면 된다.

4. Section 085

차트에 표시할 매출액(B1:B5)과 구성비(D1:D5) 처럼 계열의 값이 현저하게 차이가 날 경우에는 기본 축과 보조 축이 있는 이중 축 혼합형 차트를 사용하는 것이 가장 적합하다.

6. Section 082

데이터 요소 하나만 삭제할 수는 없으며, 데이터 요소를 선택한 후 [Del]을 누르면 해당 데이터 계열이 모두 삭제된다.

7. Section 083

데이터 요소에는 문자열을 입력할 수 없으며, 임의의 문자열 또한 삽입할 수 없다.

8. Section 085

- ① 세로 막대형 : 각 항목 간의 값을 막대의 길이로 비교·분석하는데 적합함
- ② 방사형 : 많은 데이터 계열의 집합적인 값을 나타낼 때 사용됨
- ④ 영역형 : 시간에 따른 각 값의 변화량을 비교할 때 사용되는 차트로, 전체 영역과 특정 값의 영역을 비교해 전체와 부분 간의 관계를 나타냄

9. Section 083

차트 시트의 'Chart1' 시트 탭을 [Ctrl]을 누른 채 워크시트 안 쪽으로 끌면 동일한 내용의 시트가 'Chart1(2)' 라는 이름으로 하나 더 생성된다.

10. Section 083

3차원, 방사형, 원형, 도넛형, 표면형 차트에는 추세선을 추가할 수 없다.

11. Section 083

추가할 데이터를 범위로 지정한 후 차트 영역 위로 드래그하면 차트에 데이터가 추가되는 것이 아니라 차트 아래로 데이터가 이동한다.

12. Section 083

추세선의 종류에는 선형, 로그, 다항식, 거듭제곱, 지수, 이동평균이 있다.

13. Section 083

추세선에 지정한 이름은 범례에 표시된다. ③은 '추세선 서식' 대화상자의 '예측'을, ④는 '수식을 차트에 표시'를 이용하여 설정할 수 있다.

14. Section 082

그림 영역은 가로(항목) 축과 세로(값) 축으로 둘러싸인 영역이다. 차트 전체를 의미하는 것은 차트 영역이다.

16. Section 085

오차 막대를 화면에 표시하는 방법에는 '방향'과 '끝 스타일'이 있고, '방향'에는 모두, 음의 값, 양의 값이 있으며, '끝 스타일'에는 끝 모양 없음과 끝 모양이 있다.

17. Section 085

어떤 차트에서도 차트 제목을 데이터 레이블로 설정할 수는 없다. 원형 차트에서 데이터 레이블로 설정할 수 있는 내용에는 계열 이름, 항목 이름, 값, 백분율이 있다.

18. Section 085

원형 차트에서는 첫째 조각의 각을 0도에서 360도 사이의 값을 이용하여 회전시킬 수 있다.

19. Section 085

이중(축) 차트에서 왼쪽에 나타나는 눈금이 기본 축이고, 오른쪽에 나타나는 눈금이 보조 축이다.

20. Section 085

③은 분산형 차트에 대한 설명이다.

21. Section 084

차트 제목, 범례, 축 제목 등과 같은 차트 요소들에 대해서는 맞춤, 그룹, 회전 등의 정렬 사항을 설정할 수 없다. 여기서 말하는 정렬 설정은 여러 개체를 선택했을 때 위쪽 맞춤, 아래쪽 맞춤으로 정렬하거나 여러 개체를 하나로 묶어주는 등의 기능을 말한다.

22. Section 085

거품형 차트는 데이터 계열값이 세 개, 즉 가로 축, 세로 축, 원(거품)의 크기가 될 자료가 있어야만 사용할 수 있다.

23. Section 085

- 세로 막대형 차트 : 각 항목 간의 값을 막대의 길이로 비교·분석하는 데 적합함

- 이중 축 차트 : 특정 데이터 계열의 값이 다른 데이터 계열의 값과 현저하게 차이가 날 경우나 두 가지 이상의 데이터 계열을 가진 차트에 사용함
- 원형 차트 : 전체 항목의 합에 대한 각 항목의 비율을 나타냄

24. Section 085

원형, 도넛형, 분산형, 거품형, 방사형 등의 차트에는 '데이터 표'를 나타낼 수 없다.

25. Section 085

피벗 차트 보고서에서는 분산형, 거품형, 주식형 차트를 사용할 수 없다.

26. Section 085

- ② 꺾은선형 차트는 하나의 데이터 계열을 하나의 선으로 표시한다.
- ③ 항목의 값을 선으로 표시하는 것은 꺾은선형 차트이다.
- ④ 방사형 차트는 3차원 차트로 작성할 수 없다.

4장 정답 및 해설 — 출력

1.④ 2.② 3.② 4.④ 5.③ 6.② 7.④ 8.② 9.① 10.③ 11.① 12.④ 13.② 14.①
15.③ 16.① 17.② 18.④ 19.③ 20.② 21.④ 22.①

1. Section 087

페이지 레이아웃이나 페이지 나누기 미리 보기 상태에서는 데이터를 입력하거나 수정할 수 있어도 인쇄 미리 보기 상태에서는 데이터의 입력이나 수정이 불가능하다.

2. Section 087

'머리글/바닥글'에 [날짜]를 지정하면 워크시트를 작성한 날짜가 표시되는 것이 아니라 시스템의 현재 날짜가 표시된다.

3. Section 087

수식에 입력할 때와는 달리 머리글이나 바닥글에 텍스트를 입력할 때는 큰따옴표(“)로 묶지 않고 그대로 입력하면 되고, 날짜, 시간, 페이지 번호 등을 표시하려면 표시하려는 항목명을 대괄호[]로 묶어주면 된다.

4. Section 086

현재 워크시트에 틀 고정 기능이 적용된 경우 [보기] → [창] → [틀 고정] → [틀 고정 취소]를 이용하여 틀 고정을 해제할 수 있다.

6. Section 086

화면에 표시되는 틀 고정 형태는 인쇄에 영향을 주지 않는다.

7. Section 086

화면의 확대/축소는 인쇄 시 적용되지 않는다.

8. Section 087

행/열 머리글과 눈금선을 인쇄하려면 [페이지 설정]의 '시트' 탭에서 '행/열 머리글'과 '눈금선'을 선택해야 한다. [보기] → [표시/숨기기]에서 '눈금선'과 '머리글'을 선택하거나 [Office 단추] → [Excel 옵션] → '고급' 탭의 '이 워크시트의 표시 옵션'에서 '눈금선 표시'와 '행 및 열 머리글 표시'를 선택하면 화면에만 표시되고 인쇄 시에는 출력되지 않는다.

9. Section 087

[페이지 설정]에서 수동으로 인쇄 배율을 지정할 수 있으며, 배율은 보통 크기의 10%에서 400%까지 가능하다.

10. Section 087

'페이지 설정' 대화상자에는 '일련번호로 출력'이라는 옵션이 없다.

11. Section 087

[페이지] 탭의 '자동 맞춤'에서 용지 너비와 용지 높이를 모두 1로 지정하면 전체 내용을 한 페이지에 인쇄한다.

12. Section 087

'- &[페이지 번호] -'와 같이 입력하면 '- 1 -'과 같이 해당 페이지 번호가 표시되는데, '& - [페이지번호] -'와 같이 입력하면 '- [페이지번호] -'와 같이 텍스트 내용이 그대로 표시된다.

13. Section 087

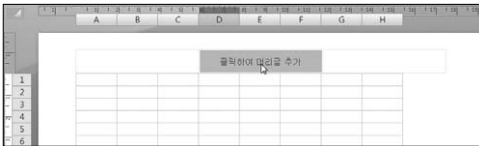
[페이지 나누기 미리 보기] 상태에서 페이지 나누기 선을 마우스로 드래그하면 셀의 크기가 조절되는 것이 아니라 페이지 나누기 선의 위치가 이동된다.

14. Section 087

[Office 단추] → [인쇄] → [인쇄 미리 보기] → [페이지 설정]을 클릭하면 '페이지 설정' 대화상자가 표시되는데, 이 대화상자의 '시트' 탭에서는 인쇄 영역, 반복할 행, 반복할 열, 메모 인쇄 여부를 설정할 수 없도록 비활성화되어 표시된다.

15. Section 087

[보기] 탭의 [통합 문서 보기] 그룹에서 머리글이나 바닥글을 쉽게 삽입하려면 '페이지 레이아웃'을 클릭한 후 워크시트 페이지의 맨 위나 맨 아래에 표시된 머리글 또는 바닥글 텍스트 상자를 클릭한 후 내용을 입력해야 한다.



16. Section 087

머리글과 바닥글에서 글자의 색을 다른색으로 변경할 수 있다.

17. Section 085

행 높이와 열 너비를 변경하면 '자동 페이지 나누기'의 위치도 변경된다.

18. Section 088

[Office 단추] → [인쇄] → [인쇄 미리 보기]를 선택한 후 '페이지 설정'을 누르면 [페이지 레이아웃] → [페이지 설정의]을 클릭한 것과 같이 '페이지 설정' 대화상자가 표시되지만, [인쇄 미리 보기] 상태에서는 인쇄 영역 설정란이 비활성화되어 영역을 설정할 수 없다.

19. Section 088

여러 시트를 인쇄하려면 Shift나 Ctrl을 이용하여 인쇄할 시트를 선택한 후 '인쇄' 대화상자에서 '선택한 시트'를 지정하고 <확인>을 눌러야 한다.

20. Section 087

- [Office 단추] → [인쇄]를 실행해 인쇄 대상에서 '전체 통합 문서'를 선택하면 통합 문서 내의 모든 워크시트를 인쇄할 수 있다.
- 인쇄 영역으로 선택한 내용이 용지의 가로/세로 방향에 대해 가운데로 위치하게 하려면 [페이지 레이아웃] → [페이지 설정의]을 클릭하여 '여백' 탭을 이용해야 한다.
- 임의의 페이지에서 인쇄를 시작하려면 '인쇄' 대화상자의 '인쇄 범위'에 인쇄할 페이지를 설정한다.

21. Section 088

삽입된 배경 그림은 인쇄할 수 없다.

22. Section 088

[페이지 레이아웃] → [페이지 설정] → [배경]을 이용하여 워크시트에 추가한 배경 무늬는 인쇄할 수 없다.

5장 정답 및 해설 — 데이터 관리

- 1.④ 2.① 3.② 4.② 5.③ 6.② 7.④ 8.④ 9.③ 10.④ 11.③ 12.② 13.④ 14.④
15.④ 16.④ 17.③ 18.④ 19.④ 20.④ 21.④ 22.② 23.① 24.② 25.④ 26.④ 27.④

1. Section 089

데이터 목록 중간에 빈 행이 있기 때문에 두 개의 데이터 목록으로 취급된다.

2. Section 089

엑셀에서 기본적으로 제공하는 사용자 지정 목록은 수정하거나 삭제할 수 없다.

3. Section 092

- ① 원본 텍스트 파일이 수정되어도 가져온 데이터에 자동으로 반영되지는 않는다. 가져온 데이터에 수정된 내용을 반영하려면 가져온 데이터의 임의의 셀을 클릭한 후 [데이터] → [연결] → [모두 새로 고침]을 클릭해야 한다.
- ③ 텍스트 파일에서 특정 열(Column)만 선택하여 가져올 수 있다.
- ④ 사용할 수 있는 텍스트 파일 형식에는 *.txt, *.prn, *.csv가 있다.

4. Section 089

내림차순으로 정렬하면 영문자가 한글의 앞이 아니라 뒤에 위치한다.

5. Section 089

셀 포인터가 표 범위 내에 있지 않을 때는 '지정한 범위를 사용하여 명령을 완료할 수 없습니다. 범위 내에서 셀 하나를 선택한 다음 명령을 다시 실행하십시오.'라는 오류 메시지가 표시된다.

6. Section 089

'데이터 연결 마법사'를 이용하면 Access, dBASE, FoxPro, Excel, Oracle, Paradox 및 SQL Server를 비롯한 다양한 종류의 데이터를 Excel로 가져올 수는 있으나 여러 테이블을 조인하여 가져올 수는 없다.

※ '데이터 연결 마법사' 실행 : [데이터] → [외부 데이터 가져오기] → [기타 원본] → [데이터 연결 마법사] 선택

7. Section 089

그림 (나)는 그림 (가)를 정렬한 것으로, '정렬 옵션'의 방향을 '왼쪽에서 오른쪽'을 선택한 후 수행한 것이다.

8. Section 089

사용자 지정 목록을 추가할 때 숫자를 첫 문자로 사용하거나 숫자로만 이루어진 사용자 지정 목록도 정렬에 적용된다.

9. Section 092

열 구분선의 위치를 옮기려면 구분선을 마우스로 드래그하여 이동하면 된다.

10. Section 090

자동 필터는 원본 데이터가 있는 위치에만 결과 데이터를 표시할 수 있다.

11. Section 090, 091

만능 문자(와일드 카드)

- 모든 문자를 대신하여 사용하는 문자
- '*'는 문자의 모든 자리를 대신할 수 있지만, '?'는 문자의 한 자리만 대신할 수 있다.

12. Section 093

외부 데이터를 쿼리로 저장하려면 '쿼리 저장'을 클릭하거나 'Microsoft Query에서 데이터 보기 또는 쿼리 편집'을 선택하면 표시되는 Microsoft Query에서 [파일] → [저장]을 선택해야 한다.

13. Section 091

- 그림은 AND와 OR 조건이 결합된 형태로, 거주지가 서울이고 연령이 25세 이하이거나 거주지가 부산이고 연령이 50세 이상인 사람을 추출하는 조건이다.
- AND와 OR 조건이 같이 있을 경우, AND 조건이 우선이다.

14. Section 091

- 고급 필터의 조건으로 수식을 입력할 경우, 조건으로 지정될 범위의 첫 행에는 아무것도 입력하지 않거나 원본 데이터의 필드의 이름과 다른 내용을 입력해야 한다.
- 전체 평균을 구하는 범위는 항상 고정적으로 사용되어야 하므로 AVERAGE 함수의 인수는 절대 참조로 지정해야 한다.

15. Section 091

조건이 서로 다른 행에 있으므로 OR 조건이다. 상품명이 '디오'로 끝나는 3자리이거나 금액이 40000원 미만인 데이터를 추출한다.

16. Section 093

새 쿼리는 하나의 통합 문서에 대해서만 만들 수 있다.

18. Section 093

[데이터] 탭 [연결] 그룹에서 [속성]을 클릭하면 나타나는 '연

결 속성' 대화상자에서는 기존 Access 파일의 연결을 변경할 수는 있어도 새로 추가하거나 제거할 수는 없다.

19. Section 093

연결을 제거해도 현재 통합 문서에 가져온 데이터는 삭제되지 않지만 데이터 새로 고침은 더 이상 실행할 수 없다.

20. Section 093

외부 데이터는 쿼리 마법사, 텍스트 마법사, 웹 쿼리 등의 방법을 이용하여 가져올 수 있다.

21. Section 093

오프라인 큐브 파일의 확장자는 cub이고, iqy는 웹 쿼리 파일의 확장자이다.

22. Section 093

'데이터 필드' 대화상자에서 일반적으로 한 열에 대해 지정할 수 있는 조건은 제한이 없다.

23. Section 093

accdb 파일을 쿼리 마법사로 가져올 때 설정할 수 있는 내용은 지정할 특정 필드 추출, 포함할 행에 대한 조건, 정렬 순서 등이다. 쿼리 마법사에 암호를 지정하는 기능은 없다.

24. Section 091

조건이 모두 서로 다른 행에 있으므로 모두 OR 조건으로 연결된다.

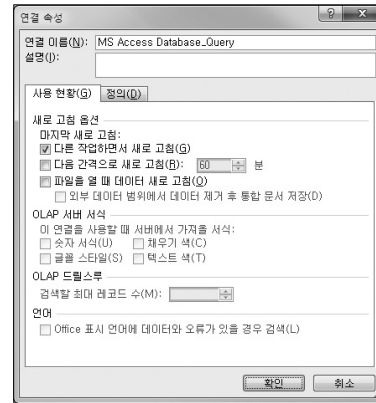
26. Section 092

- ① 데이터의 구분 기호로 탭, 세미콜론, 쉼표, 공백, 기타 문자를 설정할 수 있다.
- ② '외부 데이터 가져오기'를 이용하여 텍스트 파일을 불러오는 경우 텍스트가 삽입될 위치를 지정할 수 있으므로 따로 복사할 필요가 없다.
- ③ 원본 텍스트 파일이 수정되었을 경우 [데이터] → [연결] →  → [모두 새로 고침]을 이용하여 불러온 데이터를 원본 텍스트 파일과 동일하게 수정할 수 있다.

27. Section 093

연결 속성

- 실행 방법 : 새로 고침 외부 데이터 범위의 임의의 셀을 클릭한 후 [데이터] → [연결] → [연결]을 클릭한 후 '통합 문서 연결' 대화상자에서 <속성>클릭



6장 정답 및 해설 — 데이터 분석

- 1.③ 2.① 3.① 4.① 5.③ 6.④ 7.② 8.① 9.② 10.④ 11.③ 12.② 13.① 14.③
15.① 16.③ 17.① 18.① 19.① 20.③ 21.② 22.② 23.③ 24.② 25.③ 26.① 27.② 28.③
29.④ 30.④ 31.②

1. Section 094

부분합을 작성한 후 윤곽 기호를 눌러 특정한 데이터가 표시된 상태에서 차트를 작성하면 화면에 표시된 데이터만 차트에 표시된다.

2. Section 097

수식이 입력되어야 하는 셀은 [B3] 셀이다.

※ [B3] 셀에 입력된 '=A2*B2'는 보기 ③번과 문제에 제시된 그림을 통해 유추할 수 있다. ③번 보기에서 '행 입력 셀'에 A2, '열 입력 셀'에 B2를 지정하였으므로 수식은 반드시 이 두 셀을 이용하여 값을 계산해야 하고, [C4:F7] 영역

에 표시된 값을 통해 두 셀을 곱해 값을 구했음을 알 수 있다. [C3] 셀의 값 5%와 [B4] 셀의 값 2,000을 곱한 100이 [C4] 셀에 입력되어 있다.

3. Section 094

그룹화할 항목

그룹으로 묶을 기준이 되는 항목으로, 오름차순이나 내림차순, 사용자 지정 목록 순으로 정렬되어 있어야 한다.

5. Section 095

- ① 원본의 자료가 변경된 경우 피벗 테이블에 적용하려면 [데이터] → [연결] → [모두 새로 고침] → [모두 새로 고침]을 선택해야 한다.
- ② 피벗 테이블의 작성 장소를 지정하지 않으면 새 워크시트에 결과가 나타난다.
- ④ 피벗 테이블을 삭제하려면 피벗 테이블 전체를 범위로 지정하고 [Delete]를 누른다.

6. Section 097

사용할 레이블을 지정하지 않고 위치를 기준으로 데이터를 통합하면 원본 영역의 행 또는 열 레이블은 대상 영역으로 복사되지 않고 공백으로 표시된다.

7. Section 095

문자 필드를 그룹 지정하면 자동으로 그룹명이 지정되며, '그룹화' 대화상자는 표시되지 않는다.

8. Section 095

데이터 표 기능을 통해 입력된 셀들은 표 범위의 일부뿐만 수정할 수 없다.

9. Section 095

값 표시 형식 항목

- 행 방향의 비율 : 행을 기준으로 비율을 계산
 - [기준값]과의 차이 : 기준 항목과의 차이 계산
 - [기준값]에 대한 비율 : 기준 항목에 대한 비율 계산
 - [기준값]에 대한 비율의 차이 : 기준 항목에 대한 비율과의 차이 계산
- ※ 그림 (나)에서 프린터가 100%이므로 프린터를 기준으로 각 항목의 비율을 적용한 것이다.

11. Section 095

작성된 피벗 테이블을 삭제하면 피벗 차트는 삭제되지 않고 일반 차트로 변경되어 표시된다.

12. Section 095

'피벗 테이블 만들기' 대화상자에서 작성한 피벗 테이블의 위치를 기존 워크시트나 새 워크시트로 지정할 수 있다.

13. Section 095

	A	B	C	D
보고서 필터 필드	제품명	(모두)		
값 필드	합계 : 판매량	대리점		
	지역	A	B	총합계
행 레이블	동작구	23	39	62
	마포구	29	29	58
	서초구	22	46	68
	총합계	74	114	188
	0			

15. Section 096

시나리오는 별도의 파일로 저장되는 것이 아니라 별도의 시트로 작성된다.

16. Section 096

목표값 찾기에서는 변경할 입력값에 제한 조건을 지정할 수 없다. 변경할 입력값에 제한 조건을 지정하려면 해 찾기를 이용해야 한다.

17. Section 096

이 문제는 평균(수식 셀 : D2)이 90점(찾는 값)이 되기 위해서 수학 과목의 점수(값을 바꿀 셀 : C2)는 몇 점이 되어야 하는지를 묻는 문제이다.

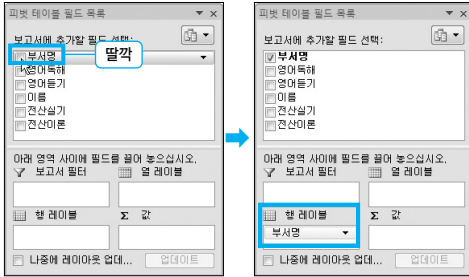
18. Section 096

평균(B29)이 수식 셀이고, 90이 찾는 값이며, 생물 성적(B26)이 값을 바꿀 셀이 된다.

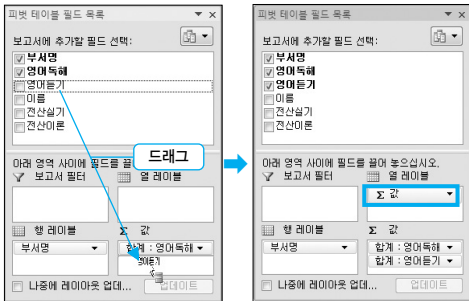
19. Section 095

피벗 테이블의 원본 데이터를 수정할 경우 수정된 내용이 피벗 테이블에 자동으로 반영되지 않는다. 원본 데이터의 수정 사항을 피벗 테이블에 반영하려면 [모두 새로 고침]을 실행해야 한다. 나머지 보기를 살펴보면 다음과 같다.

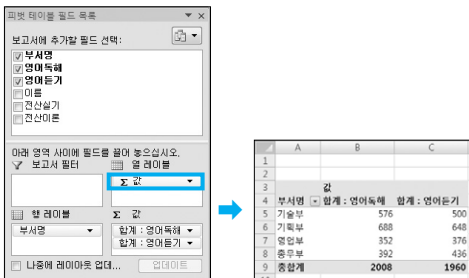
- ② '피벗 테이블 필드 목록' 창의 '보고서에 추가할 필드 선택'에서 텍스트 형식으로 된 '부서명' 필드의 체크 표시를 선택하면 '행 레이블' 영역에 '부서명' 자동으로 필드가 표시된다.



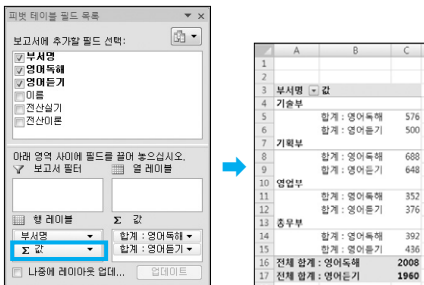
- ③ 값 영역에 '영어독해'와 '영어듣기' 필드를 지정하면 '열 레이블' 영역에 값 필드가 자동으로 표시된다.



- ④ 값 필드가 '열 레이블' 영역에 있는 경우



- 값 필드가 '행 레이블' 영역에 있는 경우



20. Section 095, 097

정렬을 가장 먼저 수행해야 하는 작업은 부분합이다.

21. Section 097

‘통합’ 대화상자의 모든 참조 영역에는 ‘범위1’과 ‘범위2’가 지정되어 있는 것으로 미루어 볼 때 항목을 기준으로 통합하려는 것이다.

22. Section 095

피벗 차트 보고서 생성시 차트 제목은 자동으로 표시되지 않는다.

23. Section 096

결과 셀에는 반드시 변경 셀을 참조하는 수식이 입력되어 있어야 한다.

25. Section 095

‘피벗 테이블 만들기’ 대화상자에서 ‘외부 데이터 원본 사용’을 선택하면 액세스 데이터베이스 파일을 이용하여 피벗 테이블을 작성할 수 있다.

27. Section 096

시나리오의 결과는 요약 보고서나 피벗 테이블 보고서로 작성할 수 있다.

28. Section 094

1을 선택하면 모든 하위 수준의 데이터를 숨기고, 전체 결과만을 표시한다.

29. Section 097

통합 시 병합된 셀이 있을 경우 올바른 결과가 표시되지 않는다. 아래와 같이 병합된 셀을 통합할 경우 병합된 셀 중 첫 번째 셀(B4)에만 값이 입력된 것으로 계산되고, 나머지(C4)는 빈 셀로 계산된다.

	A	B	C	D	E	F	G
1	상반기 판매량				하반기 판매량		
2							
3		냉장고	세탁기		냉장고	세탁기	
4	서울점	500			서울점	500	100
5	부산점	200	150		부산점	300	100
6	강원점	100	50		강원점	200	100
7							
8	총 판매량						
9							
10		냉장고	세탁기				
11	서울점	1000	100				
12	부산점	500	250				
13	강원점	300	150				

※ 세탁기 총 판매량이 600으로 표시되어야 하는데, 100만 표시된다.

30. Section 095

피벗 차트 보고서와 표준 차트 모두 그림 영역, 범례, 차트 제목, 축 제목 등을 이동하거나 그 크기를 조정할 수 있다.

31. Section 097

변화되는 값은 가중치이고, 가중치의 변경 값이 한 열(B)에 입력되어 있으므로 '데이터 표' 대화상자의 '열 입력 셀'에 가

중치를 지정하면 된다. [C3] 셀에 입력된 수식 '=D2*A2'에서 [D2] 셀에 입력된 값이 성적이므로 [A2] 셀이 가중치임을 알 수 있다. 그러므로 '데이터 표' 대화상자의 '열 입력 셀'에 'A2'를 입력하면 된다.

7장 정답 및 해설 — 매크로 작성과 VBA 프로그래밍

1.① 2.④ 3.② 4.④ 5.① 6.② 7.④ 8.③ 9.③ 10.② 11.④ 12.④ 13.① 14.①
15.② 16.① 17.④ 18.② 19.② 20.③ 21.① 22.③ 23.① 24.② 25.③ 26.② 27.① 28.①
29.① 30.④ 31.④ 32.④ 33.① 34.② 35.④ 36.① 37.④ 38.③ 39.③ 40.③ 41.② 42.①
43.②

1. Section 098, 099

자주 사용되는 매크로는 PERSON.XLSB가 아니라 PERSONAL.XLSB로 저장해야 한다.

2. Section 098

매크로 실행을 위한 바로 가기 키를 지정하지 않아도 매크로 작성이 가능하다. 도형이나 양식 버튼, 사용자 지정 단추 등에 매크로를 지정하여 실행할 수 있다.

3. Section 098

매크로를 기록하면 기본적으로 절대 참조로 기록되므로, 매크로를 실행할 때 선택한 셀이 아니라 매크로를 기록할 때 선택한 셀을 이용한다.

4. Section 101

변수를 선언하는 Dim은 선언하는 위치에 따라 해당 변수의 사용 범위가 달라진다. 프로시저의 처음에 선언하면 해당 프로시저 내에서만 사용이 가능하고, 모듈의 처음에 선언하면 해당 모듈 내의 모든 프로시저에서 사용이 가능하다. 이 문제와 같이 LocalVariable 프로시저 안에서 strMsg 변수를 선언하면 이 프로시저 안에서만 strMsg 변수를 사용할 수 있다. OutsideScopo 프로시저에서 strMsg 변수를 사용하면 LocalVariable 프로시저에서 선언된 strMsg 변수가 아닌 새로운 변수로 인식하여 메시지 박스에 아무것도 표시되지 않는다. strMsg 변수를 두 프로시저에서 모두 사용하려면 아래

와 같이 두 프로시저에서 모두 변수 선언을 해야 한다.

```
1 Sub LocalVariable()  
2   Dim strMsg As String  
3   strMsg = "작업이 완료되었습니다."  
4   MsgBox strMsg  
5 End Sub  
  
6 Sub OutsideScopo()  
7   Dim strMsg As String  
8   MsgBox strMsg  
9 End Sub
```

- 1 LocalVariable 프로시저를 시작한다.
- 2 strMsg를 문자열 변수로 선언한다.
- 3 strMsg 변수에 "작업이 완료되었습니다."를 저장한다.
- 4 strMsg 변수에 저장된 "작업이 완료되었습니다."가 표시된 메시지 박스를 표시한다.
- 5 LocalVariable 프로시저의 끝이다.
- 6 OutsideScopo 프로시저를 시작이다.
- 7 strMsg를 문자열 변수로 선언한다.
- 8 strMsg 변수에 저장된 내용이 표시된 메시지 박스를 표시한다. OutsideScopo 프로시저에서 선언한 strMsg 변수에는 아무것도 저장되어 있지 않으므로 빈 메시지가 표시된다.
- 9 OutsideScopo 프로시저의 끝이다.

5. Section 098

매크로를 실행하는 바로 가기 키는 **[Alt]+[F8]**이다.

6. Section 100

폼 모듈과 보고서 모듈은 이벤트 프로시저를 포함할 수 있다.

7. Section 098

ActiveX 컨트롤을 이용하여 도형을 작성할 경우 바로 가기 메뉴에 '매크로 지정' 메뉴가 표시되지 않는다. 즉 ActiveX 컨트롤을 이용해서 만든 도형은 '매크로 지정' 메뉴를 이용하여 매크로를 지정할 수 없다는 것이다.

8. Section 108

문제의 매크로 코드를 살펴보면 다음과 같다.

```
Sub Macro1()
```

```
1 For i = 1 To 10 Step 2
```

```
2 ActiveCell.FormulaR1C1 = i
```

```
3 Selection.Font.ColorIndex = 5
```

```
4 ActiveCell.Offset(1, 0).Range("A1").Select
```

```
5 Next i
```

```
End Sub
```

① i가 1에서 10이 될 때까지 2씩 증가하면서 ②~④번을 수행한다.

② 활성화된 셀에 i 값을 입력한다.

- FormulaR1C1 : R1C1 스타일의 수식을 지정하는 속성이다. 여기서는 단순히 i 값을 입력한다.

③ 선택한 영역의 글꼴 색을 파랑색으로 지정한다.

④ 활성화된 셀에서 아래로 1칸, 오른쪽으로 0칸 이동한 후 그 셀을 기준으로 첫 번째 열(A), 첫 번째 행(1)을 선택한다.

- Offset : 지정된 범위에서 떨어진 범위

- Range("A1") : [A1] 셀을 의미하는 것이 아니라 현재 선택된 셀을 기준으로 첫 번째 열(A), 첫 번째 행(1)을 의미한다. 'Range("A2")'로 지정하면 첫 번째 열(A), 두 번째 행(2)을 의미한다.

※ 'ActiveCell.Offset(1, 0).Select'로 작성해도 결과는 동일하다.

⑤ 반복문이 끝날 때까지 ①번으로 이동한다.

활성화된 셀이 [B2] 셀이라고 할 때 반복문 실행에 따른 변수의 변화는 아래와 같다.

실행횟수	i	ActiveCell.Offset(1, 0).Range("A1").Select
1	1	활성화된 [B2] 셀에서 1행 떨어진 셀 선택(B3)
2	3	활성화된 [B3] 셀에서 1행 떨어진 셀 선택(B4)
3	5	활성화된 [B4] 셀에서 1행 떨어진 셀 선택(B5)
4	7	활성화된 [B5] 셀에서 1행 떨어진 셀 선택(B6)
5	9	활성화된 [B6] 셀에서 1행 떨어진 셀 선택(B7)
6	11	

i가 11일면 10보다 크므로 반복문을 종료한다. 매크로 모듈을 실행하면 결과는 다음과 같이 입력된다.

	A	B	C
1			
2		1	
3		3	
4		5	
5		7	
6		9	
7			

9. Section 105

ActiveCell.Offset(2,3)은 현재 셀로부터 아래쪽으로 2행, 오른쪽으로 3열 떨어진 위치의 셀을 의미한다. 'Comment.Visible'은 메모의 항상 표시 여부를 지정하는 속성으로 'True'로 지정하면 메모를 항상 화면에 표시하고, 'False'로 지정하면 셀 포인터가 옮겨갈 때만 메모를 표시한다.

10. Section 105

워크시트의 셀이나 셀 범위를 지정하는 속성은 Range라는 것만 알아도 맞힐 수 있는 문제이다. ㉠ 부분에 들어갈 내용은 나머지 코드를 보고 'Select'라는 것을 유추할 수 있다.

문제에 제시된 코드의 의미는 다음과 같다.

```
Sub Macro1()
```

```
1 Range("A1").Select
```

```
2 ActiveCell.FormulaR1C1 = "Name"
```

```
3 Range("B1").Select
```

```
4 ActiveCell.FormulaR1C1 = "Address"
```

```
5 Range("B2").Select
```

```
End Sub
```

① [A1] 셀을 선택한다.

- Range : 워크시트의 셀이나 셀 범위
- Select : 선택

② 활성화된 셀에 "Name"을 입력한다.

- FormulaR1C1 : R1C1 스타일의 수식을 지정하는 속성이다. 여기서는 단순히 텍스트를 입력하므로, 입력할 텍스트를 큰따옴표(")로 묶어서 입력하면 된다.

- ③ [B1] 셀을 선택한다.
- ④ 활성화된 셀에 “Address”를 입력한다.
- ⑤ [B2] 셀을 선택한다.

11. Section 100

개체에 따라 발생하는 이벤트의 종류가 다르다.

12. Section 098

- ① For문을 닫는 문장이 'Next Counter'로 쓰였기 때문에 반복 변수는 'Counter'이다. For문은 'For 반복변수=시작값 To 최종값' 형식으로 입력하므로, ①에는 'For'가 입력되어야 한다.
- ② 행과 열을 구분하여 셀 주소를 지정할 수 있는 속성은 Cells이므로, ②에는 Cells를 입력해야 한다.
- ③ Cells은 'Cells(행 번호, 열 번호)' 형식으로 지정하고 데이 터는 워크시트의 [C1:C20] 영역에 입력되어 있으므로, ③에는 C열에 해당하는 열 번호인 3을 입력해야 한다.
- ④ 절대 값이 0.5보다 작은 숫자를 모두 0으로 지정하려면 ④에는 값을 절대값으로 변경하는 Abs 함수를 입력해야 한다.

지문의 프로시저에 사용된 코드를 살펴보면 다음과 같다.

```
Sub ToZero()
1 For Counter = 1 To 20
2   Set curCell=Worksheets("Sheet1").Cells(Counter, 3)
3   If Abs(curCell, Value) < 0.5 Then curCell, Value = 0
4 Next Counter
End Sub
```

- ① Counter가 1에서 20이 될 때까지 ②~③번을 반복하여 수행한다.
 - ② Sheet1 워크시트의 Cells(Counter, 3)에 입력된 값을 curCell 변수에 저장한다.
 - ③ curCell 변수의 절대값이 0.5보다 작으면 curCell 변수의 값을 0으로 치환한다.
 - ④ ①번으로 이동한다.
- ※ Cells(Counter, 3)의 의미는 Counter가 1일 때는 1행 3열, Counter가 2일 때는 2행 3열, Counter가 3일 때는 3행 3열을 의미한다. 1행 3열은 C1, 2행 3열은 C2, 3행 3열은 C3이 된다. 여기서는 Counter가 1에서 20까지 증가하므로 [C1:C20] 영역에 입력된 값 중 절대값이 0.5보다 작은 값을 0으로 변경한다.

13. Section 100

범위를 지정한 후 지정된 범위에 대해서만 매크로를 실행하도록 작성하려면 [매크로 기록]과 [매크로 실행]을 하기 전에

해당 부분을 범위로 지정해야 한다.

14. Section 098

매크로를 기록하면 Microsoft Visual Basic 모듈로 저장되므로 Visual Basic Editor를 사용하여 오류를 고치거나 매크로 내용을 변경할 수 있다.

15. Section 098

매크로를 실행할 때 현재 셀의 위치에 따라 매크로가 적용되는 셀이 달라지도록 하려면 절대 참조가 아니라 상대 참조로 매크로를 기록해야 한다.

16. Section 102

문제에 제시된 코드를 하나하나 살펴보면 아래와 같다.

```
Sub Chk()
1 LoopN = 0
2 Num = 10
3 Do
4   Num = Num - 2
5   LoopN = LoopN + 1
6 Loop While Num > 5
7 MsgBox "루프를 " & LoopN & "번 반복."
End Sub
```

- ① LoopN에 0을 치환한다.
- ② Num에 10을 치환한다.
- ③ ⑥번에 지정한 조건식에 만족할 때까지 ④~⑤번을 반복하여 수행한다.
- ④ Num 값을 2 감소시킨다.
- ⑤ LoopN 값을 1 증가시킨다.
- ⑥ Num이 5보다 클 경우 ③번을 수행하고 아니면 반복문을 벗어난다(Do~Loop While은 조건식이 참인 동안 반복 실행함).

반복문 실행에 따른 변수의 변화는 아래와 같다.

실행횟수	Num	LoopN
초기값	10	0
1	8	1
2	6	2
3	4	3

- ※ 두 번 째로 ④, ⑤번 문장을 실행을 했을 때 Num의 값은 6, 즉 5보다 크므로 다시 ③번으로 가서 세 번 째로 ④, ⑤번 문장을 실행하면 Num은 4, LoopN은 3이 되며, ⑥번 문장

에서 Num이 5보다 크냐는 조건에 걸려 반복문을 빠져나온다. 그 때의 LoopN의 값은 3이다.

17. Section 101

예약어는 변수명으로 사용할 수 없다.

18. Section 101

문제에 제시된 코드의 의미는 다음과 같다.

```
Sub Test()
    ❶ With Worksheets("Sheet1").Range("A1:A100")
    ❷ .Value = 50
    ❸ .Font.Bold = True
    ❹ End With
End Sub
```

- ❶ • With문의 시작으로 ❷~❸번 문장에 입력해야 할 Worksheets("Sheet1").Range("A1:A100")을 생략한다. With문이 없으면 ❷번은 Worksheets("Sheet1").Range("A1:A100").Value = 50' 처럼 기술해야 한다.
- Sheet1 워크시트의 [A1:A100] 셀이 범위로 지정되었음을 의미한다.
- ❷ 범위로 지정된 셀에 50을 입력한다.
- ❸ 범위로 지정된 셀의 글꼴 스타일을 '굵게'로 지정한다.
- ❹ With문의 끝이다.

19. Section 102

그림과 같이 실행되는 것은 ❷번이다. ❷번의 프로시저를 하나하나 살펴보면 아래와 같다.

```
❶ Cells(1, 1) = 1
❷ For K = 2 To 5
❸ Cells(1, K) = Cells(1, K - 1) + K
❹ Next
```

- ❶ 1행 1열, 즉 [A1] 셀에 1을 입력한다.
 - ❷ K를 2에서 5가 될 때까지 1씩 증가시키면서 매번 ❸번 문장을 수행한다.
 - ❸ 1행 K-1열에 입력된 값에 K를 더해 1행 K열에 치환한다.
 - ❹ 반복문의 끝으로서 반복문의 시작인 ❷번으로 이동한다.
- For ~ Next문 ❷~❹의 실행에 따른 변수의 변화를 보면 다음과 같다.

실행횟수	K	Cells(1, K-1) + K	Cells(1, K)
1	2	1행1열(A1) + 2	1행2열(B1) → 3
2	3	1행2열(B1) + 3	1행3열(C1) → 6
3	4	1행3열(C1) + 4	1행4열(D1) → 10
4	5	1행4열(D1) + 5	1행5열(E1) → 15
5	6		

- ※ ❶번 실행으로 인해 [A1] 셀에는 1이 입력되어 있다.
 ※ Cells(1, K)의 의미는 K가 1일 때는 1행 1열, K가 2일 때는 1행 2열, K가 3일 때는 1행 3열을 의미한다. 1행 1열은 A1, 1행 2열은 B1, 1행 3열은 C1이 된다.
 ※ 나머지 보기를 실행할 경우에는 다음 그림과 같이 실행된다.

❶

A
1
3
6
10
15

❸

A
1
3
5
7
9

❹

A	B	C	D	E
1	3	5	7	9

20. Section 101

문제에 제시된 코드의 의미는 다음과 같다.

```
Private Sub Worksheet_Activate()
    ❶ Range("B2").CurrentRegion.Select
End Sub
```

- ❶ [B2] 셀이 포함된 데이터 범위를 모두 선택한다.
 - CurrentRegion : 데이터가 있는 인접 영역의 범위
 - Select : 선택
 ∴ [B2] 셀과 인접한 [A1], [A3], [B2] 셀에 데이터가 입력되어 있으므로 이 셀들을 모두 포함하는 영역인 [A1:B3] 영역이 모두 선택된다.

21. Section 101

j의 초기값은 0이며, Offset은 지정된 범위에서 떨어진 범위를 나타내는 속성이다.

- ❶ j가 0일 때 : [A1].Offset(0,1) = 2010 → [A1]에서 1열 떨어진 곳(B1)에 2010 입력, j = 2
- ❷ j가 2일 때 : [A1].Offset(2,1) = 2010 → [A1]에서 2행, 1열 떨어진 곳(B3)에 2010 입력, j = 4
- ❸ j가 4일 때 : [A1].Offset(4,1) = 2010 → [A1]에서 4행, 1열 떨어진 곳(B5)에 2010 입력, j = 6

22. Section 102

Private로 선언된 프로시저는 현재 프로시저가 있는 모듈 내에서만 호출이 가능하다.

23. Section 102

While문은 조건이 참인 동안 실행하는 반복문이므로 'While num > TextBox1.Text' 이 부분을 'While num <= TextBox1.Text'와 같이 입력해야 한다. 문제에 제시된 코드를 살펴보면 다음과 같다.

```
Private Sub CommandButton1_Click()
1 Dim sum As Long, num As Integer
2 num = 1
3 sum = 1
4 While num <= TextBox1.Text
5     sum = sum * num
6     num = num + 1
7 Wend
8 Label3.Caption = sum
End Sub
```

- 1 sum을 4바이트 정수형 변수, num을 2바이트 정수형 변수로 선언한다.
※ Integer는 2바이트 정수형, Long은 4바이트 정수형 자료를 의미한다.
 - 2 num 변수에 1을 치환한다.
 - 3 sum 변수에 1을 치환한다.
 - 4 num 변수가 TextBox1 컨트롤에 입력된 값보다 작거나 같으면 5~7번을 수행한다.
 - 5 sum에 num을 곱한 후 그 값을 sum에 치환한다.
 - 6 num에 1을 더한 후 그 값을 num에 치환한다.
 - 7 반복문의 끝, 4번으로 제어가 이동한다.
 - 8 Label3에 sum이 가지고 있는 값을 표시한다.
- While문의 반복에 따른 변수의 변화는 다음과 같다(단, TextBox1에 입력된 값은 3임).

반복문	sum	num
	1	1
1	1(1*1)	2(1+1)
2	2(1*2)	3(2+1)
3	6(2*3)	4(3+1)
4		

num이 4면 TextBox1에 입력된 3보다 크므로 반복문을 벗어난다. 최종적으로 Label3 컨트롤에 표시되는 값은 6이다.

24. Section 108

'Offset(intRow, 0) = txtDis'는 [B3] 셀에 txtDis의 값을 입력한다. 프로시저를 하나하나 살펴보면 아래와 같다.

```
Sub Cinput_Click()
1 Dim intRow As Integer
2 intRow = Range("B1").CurrentRegion.Rows.Count
3 With Range("B1")
4     .Offset(intRow, 0) = txtDis
5     .Offset(intRow, 1) = txtBook
6     .Offset(intRow, 2) = Val(txtPri)
7 End With
8 txtDis = "": txtBook = "": txtPri = ""
End Sub
```

- 1 intRow를 정수형 변수로 선언한다.
- 2 intRow 변수에 [B1] 셀과 연결된 범위에 있는 데이터의 행수인 2를 치환한다.
 - Range : 워크시트에서 셀이나 셀 범위
 - CurrentRegion : 지정된 셀과 연결된 범위
 - Rows : 범위에 있는 데이터의 행들
 - Count : 개수
- 3 With문으로 4~6번 문장에서 입력해야 할 'Range("B1")'을 생략한다. With문이 없으면 4번은 'Range("B1").Offset(intRow, 0) = txtDis'처럼 입력해야 한다.
- 4 [B1] 셀로부터 아래쪽으로 intRow행, 오른쪽으로 0열 떨어진 위치, 즉 intRow의 값이 2이므로 2행 0열 떨어진 [B3] 셀에 txtDis의 값을 입력한다.
- 5 [B1] 셀로부터 아래쪽으로 intRow행, 오른쪽으로 1열 떨어진 위치, 즉 [C3] 셀에 txtBook의 값을 입력한다.
- 6 [B1] 셀로부터 아래쪽으로 intRow행, 오른쪽으로 2열 떨어진 위치, 즉 [D3] 셀에 txtPri의 값을 숫자로 변환하여 입력한다. txtPri의 값이 문자이면 0이 입력된다.
- 7 With문을 종료한다.
- 8 txtDis와 txtBook, txtPri 컨트롤에 빈칸을 입력한다.

25. Section 102

- ㉠, ㉡ 데이터가 첫 번째로 입력된 셀은 [A1] 셀이고, [A1] 셀을 Cells(행, 열) 형태로 지정하면 Cells(1, 1)이므로 Cells(㉠, T_Year + 1)에서 ㉠은 1이다. T_Year + 1의 결과 값이 1이 되려면 T_Year는 0이어야 하고, T_Year가 0이 되게 하려면 For 문에서 T_Year를 0부터 시작해야 하므로 ㉠은 0이다.
- ㉢ [A1] 셀에는 2012가 입력되어 있으므로 ㉢ + T_Year의 결과 값은 2012가 되어야 한다. T_Year는 0이므로 ㉢은 2012이다.

㉔ “Quarter”가 첫 번째로 입력된 셀은 [A2] 셀이고, [A2] 셀을 Cells(행, 열) 형태로 지정하면 Cells(2, 1)이므로 ㉔의 값은 2가 되면 된다. For문에서 TQ의 초기 값으로 1이 지정되어 있으므로, TQ에 1을 더해주면 2가 되므로 ㉔은 TQ + 1이다. 문제에 제시된 프로시저를 하나하나 살펴보면 다음과 같다.

```
Sub UpTable()  
1 For T_Year = 0 To 2  
2 Cells(1, T_Year + 1).Value = 2012 + T_Year  
3 Next T_Year  
4 For TQ = 1 To 2  
5 Cells(TQ + 1, 1).Value = “Quarter” & TQ  
6 Next TQ  
End Sub
```

- 1 T_Year를 0에서 2가 될 때까지 1씩 증가시키면서 매번 2번 문장을 수행한다.
- 2 2012에 T_Year를 더해 1행 ‘T_Year + 1’ 열에 치환한다.
• Value : 값 지정
- 3 반복문의 끝으로서 반복문의 시작인 1번으로 이동한다.
For ~ Next문(1~3)의 실행에 따른 변수의 변화는 다음과 같다.

실행 횟수	T_Year	Cells(1, T_Year + 1)
1	0	2012
2	1	2013
3	2	2014
4	3	

- 4 TQ를 1에서 2가 될 때까지 1씩 증가시키면서 매번 5번 문장을 수행한다.
- 5 “Quarter”와 TQ를 연결하여 ‘TQ + 1’ 행 1열에 치환한다.
- 6 반복문의 끝으로서 반복문의 시작인 4번으로 이동한다.
For ~ Next문(4~6)의 실행에 따른 변수의 변화를 보면 다음과 같다.

실행횟수	TQ	“Quarter”& TQ	Cells(TQ + 1, 1)
1	1	Quarter1	Quarter1
2	2	Quarter2	Quarter2
3	3		

26. Section 102

코드를 설명하면 다음과 같다.

```
1 Sum = 0  
2 n = 1  
3 Do Until n>=5  
4 Sum = Sum + n  
5 n=n+2  
6 Loop
```

- 1 Sum 변수에 0을 치환한다.
- 2 n 변수에 1을 치환한다.
- 3 n이 5보다 크거나 같을 때까지, 즉 n이 5보다 작으면 4~5번을 반복 수행하고, 아니면 반복문을 벗어난다(DO Until~Loop는 조건이 거짓인 동안, 즉 참이 될 때까지 반복 실행함).
- 4 Sum 값에 n을 누적시킨다.
- 5 n 값을 2 증가시킨다.
- 6 3번으로 제어가 이동된다.
반복문 실행에 따른 변수의 변화는 다음과 같다.

실행 횟수	Do Until 문 결과	Sum	n
실행 전		0	1
1	FALSE	1(0+1)	3
2	FALSE	4(1+3)	5
3	TRUE		

그러므로 Sum의 결과 값은 4가 된다.

27. Section 104

- ②는 MsgBox (“계속할까요?”, vbYesNo + vbCritical, “확인”)
- ③은 MsgBox (“계속할까요?”, vbAbortRetryIgnore + vbExclamation, “확인”)
- ④는 MsgBox (“계속할까요?”, vbOKCancel + vbInformation, “확인”)

28. Section 100

문제에 주어진 ‘매출총이익률’ 함수 인수 대화상자에 인수로 ‘매출액’과 ‘매출원가’가 표시되어 있으므로 ①, ②번에는 각각 ‘매출액’과 ‘매출원가’가 들어감을 알 수 있다. ‘매출총이익’은 ‘매출액’에서 ‘매출원가’를 차감한 금액이므로 ③에 들어갈 값은 ‘매출액-매출원가’이다. 코드를 설명하면 다음과 같다.

```
1 Function 매출총이익률(매출액, 매출원가)  
2 매출총이익 = 매출액-매출원가  
3 매출총이익률 = Format(매출총이익/매출액, “0.00%”)  
4 End Function
```

- ①, ④ • 함수의 시작(Function)과 끝(End Function)을 의미한다.
- 매출총이익률(매출액, 매출원가) : '매출총이익률'은 사용자의 함수명이고 괄호 안의 '매출액'과 '매출원가'는 사용자로부터 입력받아 수식에 적용할 인수들이다.
- ② '매출액-매출원가'의 결과값을 '매출총이익'에 치환한다.
- ③ • '매출총이익/매출액'을 "0.00%" 형식으로 표시한다.
- Format : 데이터의 표시 형식을 지정하는 함수로, 표시 형식의 지정 방법은 사용자 지정 형식과 동일하다.

29. Section 102

- Sub ~ End Sub : 프로시저 내에 작성된 코드를 실행하는 가장 일반적인 형태로, 작성된 Sub 프로시저는 워크시트에서 매크로처럼 연결하여 사용할 수 있음
- Select ~ Case : 조건이 여러 개일 경우 각 조건별로 지정한 실행문을 사용하여 조건을 처리함
- With ~ End With : With문을 사용하면 대상 개체의 이름을 다시 참조하지 않고도 지정된 개체에 대한 여러 개의 명령문을 실행할 수 있음

30. Section 104

- vbCritical : '중대 메시지(☒)' 아이콘을 나타냄
- vbExclamation : '메시지 경고(⚠)' 아이콘을 나타냄
- vbInformation : '메시지 정보(ℹ)' 아이콘을 나타냄

31. Section 105

워크시트의 셀이나 셀 범위를 선택하는 속성은 Range이며, Range를 이용하여 특정 셀을 지정할 때는 셀 주소를 큰 따옴표(")로 묶어서 지정한다. 프로시저의 코드를 살펴보면 아래와 같다.

```
Sub profit_view()
  ① For I = 1 To 7
    ② Set cell = ActiveSheet.Range("B1").Offset(i, 0)
    ③ If cell >= 2000000 Then
      ④ cell.Interior.ColorIndex = 3 '빨강색
    Else
      ⑤ cell.Interior.ColorIndex = 5 '파랑색
    End If
    ⑥ Next I
End Sub
```

- ① i가 1에서 7이 될 때까지 ②~⑤번을 반복하여 수행한다.
- ② 활성화된 워크시트의 [B1] 셀을 기준으로 I행, 0열만큼 떨어진 셀(Offset(i, 0))을 개체 변수 cell에 저장한다. I가 1일 때는 [B2] 셀이 저장된다.

- ③ cell의 값이 2,000,000 이상이면 ④번을 수행하고, 아니면 ⑤번을 수행한다.
 - ④ cell 개체의 채우기(속성)를 빨강색으로 지정한다. 작은 따옴표(') 뒤에 있는 내용은 주석으로 실행되지 않는다.
 - ⑤ cell 개체의 채우기(속성)를 파랑색으로 지정한다.
 - ⑥ 반복문이 끝날 때까지 ①번으로 이동한다.
- ※ 반복문 실행에 따른 변수의 변화는 아래와 같다.

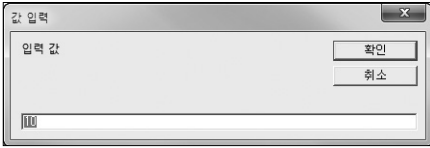
실행 횟수	I	ActiveSheet.Range("B1").Offset(i, 0)
1	1	ActiveSheet.Range("B1").Offset(1,0) → 활성화된 워크시트의 [B1]에서 1행 떨어진 곳(B2)
2	2	ActiveSheet.Range("B1").Offset(2,0) → 활성화된 워크시트의 [B1]에서 2행 떨어진 곳(B3)
3	3	ActiveSheet.Range("B1").Offset(3,0) → 활성화된 워크시트의 [B1]에서 3행 떨어진 곳(B4)
4	4	ActiveSheet.Range("B1").Offset(4,0) → 활성화된 워크시트의 [B1]에서 4행 떨어진 곳(B5)
5	5	ActiveSheet.Range("B1").Offset(5,0) → 활성화된 워크시트의 [B1]에서 5행 떨어진 곳(B6)
6	6	ActiveSheet.Range("B1").Offset(6,0) → 활성화된 워크시트의 [B1]에서 6행 떨어진 곳(B7)
7	7	ActiveSheet.Range("B1").Offset(7,0) → 활성화된 워크시트의 [B1]에서 7행 떨어진 곳(B8)
8	8	

32. Section 102

문제의 코드를 설명하면 다음과 같다.

```
Sub Macro1()
  ① for var1 = 1 To 10 Step 3
    ② su = InputBox("입력 값", "값 입력", 10)
    ③ hap = hap + Val(su)
    ④ Next
    ⑤ Cells(1, 1) = hap
  End Sub
```

- ① var1을 1에서 10이 될 때까지 3씩 증가시키면서 ②~③번을 반복하여 수행한다.
- ② 메시지 "입력 값", 대화상자 타이틀 "값 입력", 기본값이 10이 입력된 아래와 같은 대화상자를 표시한 후 입력 받은 값을 su 변수에 치환한다.



- ③ su의 값을 숫자값으로 변환한 후 hap과 더한 다음 hap에 치환한다.
 - ④ 반복문의 끝으로서, ①번으로 이동한다.
 - ⑤ hap를 1행 1열, 즉 [A1] 셀에 입력한다.
- 메시지 박스로 1, 2, 3, 4를 입력 받았을 경우의 반복문 실행에 따른 변수의 변화는 아래와 같다.

실행 횟수	var1	su	hap
1	1	1	1(0+1)
2	4	2	3(1+2)
3	7	3	6(3+3)
4	10	4	10(6+4)
5	13		

var1이 13이면 반복문을 빠져나오고, 그때의 hap은 10이다. 다른 보기가 틀린 이유는 다음과 같다.

- ① '값 입력' 대화상자는 4번 나온다.
- ② 모듈을 실행시킨 워크시트의 Cells(1,1), 즉 1행 1열인 [A1] 셀에 hap의 값이 표시된다.
- ③ '입력' 대화상자에 입력한 4개의 값에 대한 합계가 계산된다.

33. Section 105

문제의 코드를 설명하면 다음과 같다.

```

1 Private Sub Workbook_SheetActivate(ByVal Sh As
  Object)
2     MsgBox Sh.Name
End Sub

```

- ① 워크시트가 활성화 될 때마다 ②번을 실행한다. 인수로 받은 Sh는 활성화된 워크시트를 의미한다.
- ② 활성화된 워크시트(Sh)의 이름을 메시지 상자에 표시한다.

34. Section 102

문제에 주어진 사용자 정의 함수의 코드를 살펴보면 다음과 같다.

```

1 Function UserExtraction(t As Range) As Long
2     Dim i As Integer
3     Dim Tmp As String
4     If Len(t) = 0 Then UserExtraction = "": Exit
      Function
5     For i = 1 To Len(t)
6         If IsNumeric(Mid(t, i, 1)) Then
7             Tmp = Tmp & Mid(t, i, 1)
8         End If
9     Next
10    UserExtraction = Tmp
End Function

```

- ① 함수의 시작(Function)을 의미하고, 사용자 정의 함수 UserExtraction을 숫자형(4바이트)으로 선언한다. 함수명 (UserExtraction) 다음의 괄호 안에는 사용자로부터 입력 받아 수식에 적용할 인수를 지정하는 곳으로 't As Range'는 사용자가 선택(Range)한 셀의 값을 t 변수로 입력 받는다는 의미이다.
 - ② i를 숫자형(2바이트) 변수로 선언한다.
 - ③ Tmp를 문자형 변수로 선언한다.
 - ④ t 변수에 들어 있는 데이터의 길이가 0이면 UserExtraction 함수에 공백을 입력하고 함수를 끝낸다.
 - Len(문자열) : 문자열의 길이를 반환한다.
 - ⑤ i를 1부터 t 변수에 들어 있는 데이터의 길이가 될 때까지 ⑥ ~ ⑧번을 반복하여 수행한다.
 - ⑥ t 변수에 들어 있는 데이터의 i번째 글자가 숫자이면 ⑦번을 수행하고, 그렇지 않으면 ⑧번으로 이동한다.
 - IsNumeric(인수) : 인수가 숫자면 TRUE, 아니면 FALSE를 반환한다.
 - MID(문자열, 시작값, 자릿수) : 문자열의 시작 위치에서 주어진 자릿수만큼 추출한다.
 - ⑦ Tmp에 Tmp와 t 변수에 들어 있는 데이터의 i번째 글자를 연결하여 치환한다.
 - ⑧ If문을 종료한다.
 - ⑨ 반복문이 끝날 때까지 ⑤번으로 이동한다.
 - ⑩ UserExtraction에 Tmp를 치환한다.
- ※ 반복문 실행에 따른 변수의 변화는 아래와 같다.

반복 횟수	Len(t)	i	Mid(t, i, 1)	tmp
1	12	1	2	2
2		2	0	20
3		3	0	200
4		4	5	2005
5		5	년	2005
6		6	공백	2005
7		7	컴	2005
8		8	활	2005
9		9	시	2005
10		10	험	2005
11		11	1	20051
12		12	급	20051
13		13		

35. Section 103

이벤트 프로시저도 다른 프로시저에서 호출할 수 있다.

36. Section 105

Cell(2,2)는 2행 2열, 즉 [B2] 셀을 의미한다. 제시된 코드는 [B2] 셀의 값에 2를 곱하여 [B2] 셀에 치환하라는 내용이다.

37. Section 105

BeforeSave는 통합 문서가 저장되기 전에 발생하는 것으로 내용을 저장하지 않고 통합 문서를 닫으려면 'BeforeSave=True'로, 내용을 저장한 후 통합 문서를 닫으려면 'BeforeSave=False'로 지정하면 된다.

38. Section 102

③번은 1에서 9까지 합을 구하는 VBA 모듈이다. 보기의 각 코드를 살펴보면 다음과 같다.

①

```

no = 0
sum = 0
1 Do While no <= 10
2   sum = sum + no
3   no = no + 1
4 Loop
5 MsgBox sum

```

- 1 no의 값이 10보다 작거나 같으면 2~3번을 수행하고 아니면 반복문을 벗어난다.
- 2 sum에 no를 누적시킨다.

- 3 no 값을 1 증가시킨다.
- 4 1번으로 제어기가 이동된다.
- 5 sum의 값을 표시한 메시지 박스를 실행한다.

②

```

no = 0
sum = 0
1 Do
2   sum = sum + no
3   no = no + 1
4 Loop While no <= 10
MsgBox sum

```

- 1 4 번에서 지정한 조건식이 만족할 때까지 2~3번을 실행한다.
- 4 no의 값이 10보다 작거나 같으면 1번으로 이동하고 아니면 반복문을 벗어난다.

③

```

no = 0
sum = 0
1 Do While no < 10
2   sum = sum + no
3   no = no + 1
4 Loop
MsgBox sum

```

- 1 no의 값이 10보다 작으면 2~3번을 수행하고 아니면 반복문을 벗어난다.
- ※ no의 값이 10보다 작을 때까지만 2~3번을 수행하므로 1~9까지의 합계를 구한다.

④

```

sum = 0
1 For no = 1 To 10
2   sum = sum + no
3 Next
MsgBox sum

```

- 1 no를 1에서부터 10이 될 때까지 1씩 증가하면서 2번을 반복 수행한다.

39. Section 102

'Do ... Loop Until' 문은 실행문을 실행한 후 조건을 비교하는 제어문으로, 조건의 만족 여부와 상관없이 한번은 반드시 명령문을 실행하게 된다.

40. Section 105

Add는 새로운 워크시트를 삽입하는 메서드이고, 'Sheets(1)'은 첫 번째 시트를 의미하는 것으로, 'Worksheets.Add After:=Sheets(1)'로 지정하면 첫 번째 시트 뒤에 새로운 시트가 삽입되고, 'Worksheets.Add Before:=Sheets(1)'로 지정하면 첫 번째 시트 앞에 새로운 시트가 삽입된다.

41. Section 105

지문의 프로시저 내용을 살펴보면 다음과 같다.

```
Sub SumMarco()  
① Dim MyRang As range, myCell As range, MySum  
  As Integer  
② Set MyRange =  
  Worksheets("MySheet").Range("A1:A10")  
③ For Each MyCell In MyRange  
④   MySum = MySum + MyCell  
⑤ Next  
⑥ MsgBox "합계:" & MySum  
End Sub
```

- ① 셀 범위를 기억할 수 있는 Range(범위) 유형의 변수 MyRang와 MyCell, 정수형 변수 MySum을 선언한다.
- ② MySheet 워크시트의 셀 범위 [A1:A10]을 MyRang 변수에 저장한다.
- ③ MyRange 변수에 저장된 개체를 MyCell에 차례로 저장하면서 개체 수만큼 ④번을 반복 수행한다.
- ④ MySum 변수에 MyCell 변수의 값을 더해 MySum 변수에 치환한다.
- ⑤ ③번으로 이동한다.
- ⑥ 메시지 상자에 "합계:"와 MySum 변수에 저장된 값을 표시한다.

42. Section 105

Range("B1", "B5")는 Rang("B1:B5")와 마찬가지로 B1:B5 영역을 선택하므로 'Range("B1", "B5").Value = 10'을 지정하면 [B1] 셀에서 [B5] 셀까지 모든 셀에 10이 입력된다. [B1] 셀과 [B5] 셀에만 10을 입력하려면 'Range("B1, B5").Value = 10'으로 지정하면 된다.

- Range : 워크시트의 셀이나 셀 범위
- Value : 값 지정

43. Section 105

지정된 셀의 값을 의미하는 속성은 Value이므로 ②번과 같이 입력하면 [C1] 셀에 입력된 수식의 결과가 [C1]셀에 입력된다. 즉 'Range("C1").Value = Range("C1").Value'는 '=' 오른쪽의 'Range("C1"). Value'에 의해 [C1] 셀에 입력된 수식 '=A1*B1'의 결과가 계산되고 그 결과가 '=' 왼쪽의 'Range("C1"). Value'에 저장된다.



1장 정답 및 해설 — 데이터베이스 개요

1. ② 2. ④ 3. ① 4. ③ 5. ① 6. ③ 7. ② 8. ① 9. ③ 10. ③ 11. ③ 12. ③ 13. ④ 14. ②
 15. ④ 16. ② 17. ④ 18. ② 19. ① 20. ② 21. ③ 22. ③ 23. ③ 24. ④ 25. ① 26. ① 27. ② 28. ④
 29. ① 30. ① 31. ① 32. ② 33. ① 34. ① 35. ① 36. ① 37. ③ 38. ①

1. Section 109

데이터베이스에서 자료의 중앙 통제 시 자료를 한 곳에 통합하여 관리하므로 자료의 일관성 유지가 용이하다.

2. Section 110

- 개념 스키마는 기관이나 조직 전체의 관점에서 표현하는 스키마로, 데이터베이스 관리자(DBA)에 의해 구성된다.
- ④는 외부 스키마에 대한 설명이다.

3. Section 110

데이터 사전(Data Dictionary)

- 데이터베이스에 저장되어 있는 모든 데이터 개체들에 대한 정보를 유지·관리하는 시스템으로, 시스템 카탈로그(System Catalog)라고도 한다.
- 데이터 사전은 시스템 데이터베이스(System Database)에 해당한다.

4. Section 110

- 데이터 정의어(DDL) : 데이터베이스를 생성하거나 수정하는 데 사용되는 언어
- 데이터 조작어(DML) : 사용자가 응용 프로그램을 통하여 데이터베이스에 저장된 데이터를 실질적으로 처리하는 데 사용되는 언어
- 데이터 제어어(DCL) : 데이터 보안, 무결성, 데이터 회복, 병행 제어 등을 정의하는 데 사용되는 언어

5. Section 110

DBA는 데이터 제어어(DCL)를 이용하여 데이터베이스의 무결성을 유지한다.

6. Section 110

데이터 모델링을 수행하는 것은 사람인 DBA의 역할이고,

DBMS의 데이터 언어를 사용하여 데이터베이스를 정의하지만 DBMS가 데이터 언어는 아니다.

7. Section 109

데이터의 독립성

- 논리적 독립성 : 응용 프로그램과 데이터베이스를 독립시킴으로써, 데이터의 논리적 구조가 변경되더라도 응용 프로그램은 변경되지 않음
- 물리적 독립성 : 응용 프로그램과 보조기억장치 같은 물리적 장치를 독립시킴으로써, 데이터베이스 시스템의 성능 향상을 위해 새로운 디스크를 도입하더라도 응용 프로그램에는 영향을 주지 않고 데이터의 물리적 구조만 변경함
- ※ ①, ③, ④는 논리적 독립성에 대한 내용이다.

8. Section 110

외부 스키마(External Schema) = 서브 스키마(Sub Schema)

- 일반 사용자나 응용 프로그래머의 관점에서 본 스키마이다.
- 전체 데이터베이스 중 특정 응용 프로그램이나 사용자와 관련된 데이터베이스의 부분만 정의한 것이다.
- 하나의 데이터베이스 시스템에는 여러 개의 외부 스키마가 존재할 수 있으며, 하나의 외부 스키마를 여러 개의 응용 프로그램이나 사용자가 공유할 수도 있다.

9. Section 110

보안성 검사, 무결성 검사와 같은 추가적인 특징을 포함하는 것은 개념 스키마이다.

10. Section 110

데이터의 검색, 갱신, 삽입, 삭제 등이 체계적으로 수행되도록 데이터 접근 수단을 정의하는 기능은 조작 기능이다.

11. Section 110

데이터 조작용의 조건

- 사용하기 쉽고 자연 언어에 가까워야 한다.
- 데이터에 대한 연산뿐만 아니라 뷰 내의 데이터나 데이터 간의 관계를 정확하고 완전하게 명시할 수 있어야 한다.
- 데이터 언어의 효율적인 구현을 지원해야 한다. 즉 데이터 언어의 구문이 DBMS가 제공하는 기본적인 연산과 관련을 갖도록 해야 한다.

12. Section 110

데이터 정의어(DDL ; Data Definition Language)

- 데이터베이스를 생성하거나 수정하는 데 사용되는 언어
- 데이터베이스 관리자나 데이터베이스 설계자가 주로 사용한다.

13. Section 110

④는 데이터 조작용(DML)이고, 나머지는 데이터 정의어(DDL)이다.

14. Section 109

동일한 데이터가 여러 파일에 중복되었다고 해서 읽기 전용 트랜잭션에 대한 데이터의 가용도(사용하는 능력)가 감소되는 것은 아니다. 동일한 데이터가 여러 곳에 중복되어 있어도 한 개만 읽어서 사용하면 되기 때문이다.

15. Section 120

데이터베이스 관리 시스템(DBMS)은 처리 속도가 늦어질 수 있다는 단점을 가지고 있다.

16. Section 113

데이터 모델은 데이터의 개념적 · 논리적 구조를 표현하기 위한 도구로 사용되지만, 물리적 구조를 표현하지는 않는다.

17. Section 114

E-R 모델은 개체 간의 관계를 개념적으로 표현하는 방법으로, 특정 데이터베이스에 종속되지 않고 독립적인 형태로 데이터를 표현하기 위해서 사용한다.

18. Section 113

타원은 속성(Attribute)을 나타낸다.

19. Section 113

데이터 모델

개념적 데이터 모델	논리적 데이터 모델
E-R 모델	관계 모델, 계층 모델, 네트워크 모델

20. Section 114

개체 관계도(ERD)는 개념적 설계 단계에서 작성한다.

21. Section 113

사원별 거주지는 대부분 한 곳이므로, 일 대 일 관계라고 할 수 있다.

22. Section 114

하나의 도시(=1)에 다수의 사람(=N)이 거주하는 형태이다.

23. Section 112

- 차수(Degree) : 속성(Attribute)의 개수 = 4
- 카디널리티(Cardinality) : 튜플(Tuple)의 개수 = 3

24. Section 112

계층 데이터 모델은 개체 간의 관계를 부모와 자식 간의 관계로 표현하며, 네트워크 데이터 모델은 오너와 멤버(Owner-Member) 간의 관계로 표현한다.

25. Section 112

두 개 이상의 필드를 묶어서 기본키로 설정할 수 있다.

26. Section 112

문제에 제시된 테이블의 튜플(레코드) 수는 4이다.

27. Section 113

후보키(Candidate Key)

테이블을 구성하는 속성들 중에서 튜플을 유일하게 구별하기 위해 사용하는 속성들의 부분집합, 즉 기본키로 사용할 수 있는 속성들을 말한다. 후보키는 다음의 특징을 만족시켜야 하다.

- 유일성(Unique) : 하나의 키 값으로 하나의 튜플만을 유일하게 식별할 수 있어야 함
- 최소성, 비중복성 : 키를 구성하는 속성 하나를 제거하면 유일한 식별성이 파괴되도록 꼭 필요한 속성만 키로 구성해야 함

※ 후보키에서 기본키(Primary Key)를 제외한 속성들을 대

체키(Alternate Key)라 하고, 여러 속성을 조합해서 만든 기본키를 슈퍼키라고 한다.

28. Section 114

한 학과에 학생들이 여러 명 대응되고, 한 학생은 한 학과에 대응되므로 학과와 학생은 일 대 다의 관계이다. 이렇게 일 대 다의 관계일 때 '일'의 기본키와 같은 '다'의 필드가 외래키이다.

29. Section 110

- 데이터 정의어(DDL) : 데이터베이스를 생성하거나 수정하는 데 사용하는 언어
- 데이터 조작어(DML) : 사용자가 응용 프로그램을 통하여 데이터베이스에 저장된 데이터를 실질적으로 처리하는 데 사용되는 언어
- 데이터 제어어(DCL) : 데이터 보안, 무결성, 데이터 회복, 병행 제어 등을 위해 사용되는 언어

30. Section 113

이상(Anomaly) 현상

- 이상 현상이란 데이터베이스 내에 데이터들이 불필요하게 중복 저장되어, 릴레이션 조작 시 예기치 못한 현상이 발생하는 것을 말한다.
- 이상 현상에는 삽입, 삭제, 갱신 이상이 있다.

31. Section 113

정규화(Normalization)의 목적 중 하나는 릴레이션에서의 삽입, 삭제, 갱신 이상(Anomaly)을 방지하는 것이다.

32. Section 111

- 사각형 : 개체(Entity) 타입
- 다이아몬드 : 관계(Relationship) 타입
- 타원 : 속성(Attribute) 타입
- 밑줄 타원 : 기본키 속성
- 복수 타원 : 복합 속성

33. Section 112

②번은 기본키, ③번은 인덱스 설정에 대한 설명이다.

34. Section 113

하나의 레코드를 삽입할 때, 기본키의 정보가 없어서 삽입하지 못하는 경우, 즉 삽입 이상이 발생한 경우는 제 1정규화에 문제가 있는 것이다.

35. Section 110

CREATE, DROP, ALTER는 데이터 정의어이다.

36. Section 110

데이터베이스 매니저(DBM, DataBase Manager)의 기능

- 파일 매니저와의 교신
 - 데이터에 대한 실질적인 저장, 검색 및 갱신에 책임
 - 무결성 유지
 - 데이터베이스에 대한 갱신 요구가 제약 조건에 위배되는지를 결정하여 위배 시 적절한 조치를 취함
 - 비밀 보호 유지
 - 예비 조치 및 회복
 - 병행 수행 제어
- ※ 무결성을 위한 제약 조건은 데이터베이스 관리자(DBA)가 지정한다.

37. Section 110

데이터베이스를 생성하거나 수정하는데 사용되는 데이터베이스 언어는 데이터 제어어(DCL)가 아니라 데이터 정의어(DDL)이다.

38. Section 111

테이블에서 행은 레코드로서 여러 속성들의 묶음을 나타내고, 열은 하나의 레코드에 속한 각각의 속성들을 나타낸다.

2장 정답 및 해설 — 테이블(Table) 작성

1.① 2.③ 3.④ 4.④ 5.② 6.③ 7.② 8.① 9.④ 10.③ 11.③ 12.③ 13.④ 14.④
15.② 16.② 17.③ 18.① 19.④ 20.③ 21.④ 22.① 23.① 24.② 25.④ 26.② 27.② 28.③
29.③ 30.③ 31.④ 32.③ 33.④ 34.② 35.④ 36.② 37.② 38.② 39.② 40.④ 41.③

1. Section 118

정수(Long)는 4바이트, 날짜/시간과 통화 형식은 8바이트, 복제 ID는 16바이트이다.

2. Section 119

‘예/아니요’ 형식은 참/거짓, Yes/No 등과 같이 두 값 중 하나만을 선택하여 입력할 경우에 사용하는데, 학과는 종류가 다양하므로 학과의 데이터 형식으로 ‘예/아니요’는 부적합하다.

3. Section 124

레코드 추가 시 입력되는 기본값을 사용자 임의로 지정하려면 ‘기본값’ 속성을 사용해야 한다.

4. Section 117

테이블은 디자인 보기, 데이터시트 보기, 테이블 서식 파일 등을 통해 작성할 수 있으며, 테이블 미리 보기 기능은 제공되지 않는다.

5. Section 118

날짜 데이터

- 날짜 데이터도 계산은 가능하나 숫자 형식의 필드에 날짜 형식의 데이터를 입력하여 계산을 수행하기는 어렵다.
- 날짜 데이터는 날짜/시간 형식으로 지정하여 사용한다.

6. Section 118

일련 번호 형식에서 자동으로 부여된 데이터가 삭제되었을 경우 다시 부여되지 않고 다음 데이터가 부여된다.

7. Section 118

- 하이퍼링크 형식은 UNC 경로나 URL 등의 형태로 입력되는 홈페이지나 메일 주소 등의 데이터를 입력할 때 사용한다.
- 집 주소는 텍스트 형식을 사용한다.

8. Section 124

- <MEMBER> 테이블의 ID 필드는 기본 키로 지정되어 있으므로 값이 반드시 입력되어야 한다.
- ①번 구문을 실행하면 이름과 나이만 추가되므로, 기본 키는 Null 값을 가질 수 없다는 오류 메시지가 표시된다.

9. Section 119

텍스트 형식은 255자까지 입력할 수 있고, 메모 형식은 최대 65,535자까지 입력할 수 있으므로, 메모 형식을 텍스트 형식으로 변경하면 해당 데이터가 손실될 수 있다.

10. Section 123

문제의 그림에 표시된 열 너비 ‘1cm;1.501cm’를 ‘0cm;1.501cm’와 같이 수정하면 첫 번째 열의 너비는 ‘0cm’로, 두 번째 열의 너비는 ‘1.501cm’로 지정된다. 열의 너비를 ‘0cm’로 지정하면 해당하는 열이 화면에 표시되지 않으므로 3번 보기 그림 같이 두 번째 열만 표시된다. 하지만 필드에 저장되는 값은 ‘바운드 열’ 속성에 지정된 열이므로 첫 번째 열의 값 1, 2, 3, 4 중 하나가 된다.

11. Section 122

- ① = “T??” : ‘T’로 시작하는 세 글자를 찾으려는 의도였지만 Like 대신 ‘=’을 사용했으므로 ‘T??’와 정확히 일치하는 데이터만 입력할 수 있다.
- ② = “T” : “T”와 정확히 일치하는 데이터만 입력할 수 있다.
- ④ Like “T??” : T로 시작하는 3개의 문자를 입력할 수 있다.

12. Section 120

- dddd는 완전한 요일 이름을 표시하는 것으로, Friday가 표시된다.
- ‘q’는 날짜를 분기(1~4)로 나타낸다.

13. Section 120

- ①, ②, ③은 Yes 형식으로, ④는 No 형식으로 표시된다.

- 0(False)이 아닌 모든 데이터는 Yes 형식으로 표시된다.

14. Section 121

- 입력 마스크는 입력할 때의 형식을 지정하는 것으로, 데이터를 입력하거나 추가할 때 사용되며, 형식 속성은 데이터를 저장하거나 표시할 때 사용된다.
- 입력 마스크와 형식 속성이 모두 지정된 경우, 출력될 때는 형식 속성에 따른다.

15. Section 121

: 숫자나 공백을 입력하되 입력하지 않아도 무관한 기호

16. Section 121

'LA09#' 에서 L은 영문자와 한글, A는 영문자, 숫자, 한글, 0은 0~9까지의 숫자, 9와 #은 숫자나 공백을 입력할 수 있다.

17. Section 122

기본값은 새 레코드가 만들어질 때 필드에 자동으로 입력되는 값이므로, 기존의 데이터에는 아무런 영향을 주지 못한다.

18. Section 122

'Between 2000 And 30000' 은 2000에서 30000 사이의 숫자만 입력받을 수 있는 조건이므로, 입력 데이터를 30000으로 입력할 경우에는 정상적으로 입력된다.

19. Section 122

필수 속성은 필드에 데이터를 반드시 입력해야할 때 설정하는 것으로, 데이터가 입력되지 않으면 문제와 같은 메시지가 표시된다.

20. Section 122

캡션을 지정하면 디자인 보기가 아닌 데이터시트 보기에서 기존 필드명이 캡션에 설정된 내용으로 변경된다.

21. Section 123

콤보 상자의 조회 속성 중 '목록 값만 허용' 을 '아니요' 로 설정하면 목록값 이외의 데이터도 입력할 수 있다.

22. Section 123

열 개수는 2개이고, 행 수는 4개이다.

23. Section 124

기본키로 설정된 필드에는 중복된 값이나 Null 값을 입력할 수 없으므로, 입력되지 않는 경우도 있는 '도서번호' 필드는 기본키로 설정할 수 없다. 나머지 보기가 옳은 이유는 다음과 같다.

- ② '도서명' 필드는 항상 입력되어야 하므로 '필수' 속성을 지정한다.
- ③ '출판사코드' 필드는 <출판사> 테이블의 '출판사' 필드의 외래키이므로 <출판사> 테이블의 '출판사' 필드와 데이터 형식이 호환되거나 동일해야 한다.
- ④ '대여료' 필드에는 금액이 입력되어야 하므로 숫자나 통화 형식으로 지정한다.

24. Section 124

인덱스를 설정하면 데이터 검색, 정렬 등의 작업 시간은 빨라지지만 데이터 추가나 변경 시 갱신(업데이트) 속도는 느려진다.

25. Section 124

인덱스의 목적은 빠른 검색이므로 가급적 중복된 값이 없는 필드를 인덱스로 지정하는 것이 좋다.

26. Section 118

텍스트는 최대 255자, OLE 개체는 기본 1GB, 메모는 최대 65,535자를 저장할 수 있다.

27. Section 126

기본 테이블의 기본키와 관계가 설정된 테이블의 관련 필드명이 다르더라도 ①, ③, ④를 만족하면 참조 무결성을 설정할 수 있다.

28. Section 118

필드의 형식이 바이트이므로 255까지 입력할 수 있지만 유효성 검사 규칙(<=200)으로 인해 200을 초과하는 값은 입력할 수 없다.

29. Section 126

관계 설정 시 사용자가 실수로 데이터를 변경하거나 삭제하지 못하도록 '항상 참조 무결성 유지' 를 지정했기 때문에 문제의 메시지가 표시된 것이다. 이럴 경우에는 참조되는 테이블(분류)의 변경 사항이 참조하는 테이블(제품)에 자동으로 반영되도록 '관계 편집' 창의 '관련 필드 모두 업데이트' 를 선택하면 된다. 이 기능을 이용해야만 정확한 데이터 관리가 가능하다.

30. Section 127

레코드를 삽입하려면 [Ctrl] + [+], 레코드를 삭제하려면 [Ctrl] + [-] 또는 [Del]을 눌러야 한다.

31. Section 128

‘찾기’ 및 ‘바꾸기’ 대화상자에서는 ‘필드의 일부’, ‘전체 필드 일치’, ‘필드의 시작’ 이렇게 세 가지의 찾을 조건을 제공한다.

32. Section 124

관계 설정 창에서 ‘관련 필드 모두 업데이트’가 선택되어 있을 경우, 외래 키에 의해 참조되고 있는 기본 키 필드의 값이 변경되면 참조되는 외래 키 필드의 값이 모두 자동으로 변경된다.

33. Section 129

액세스에서 가져오거나 연결할 수 있는 형식에는 Access 데이터베이스의 각 개체, dBASE, Paradox, 스프레드시트, Lotus 1-2-3, Excel 파일, 텍스트 파일, Microsoft Exchange, HTML 파일 등이 있다.

34. Section 130

테이블에서는 데이터와 테이블 구조는 내보낼 수 있지만 제약 조건, 관계, 인덱스 같은 속성은 내보낼 수 없다.

35. Section 122

필드에 입력되는 숫자의 소수점 자릿수는 ‘소수 자릿수’ 속성에서 지정해야 한다.

36. Section 123

- 열 너비의 속성을 0cm로 지정하면 화면에 해당 열이 표시되지 않으므로, 0cm;3cm로 지정할 경우 첫째 열이 표시되지 않는다.
- 바운드 열을 설정하려면 ‘바운드 열’ 속성에서 지정해야 한다.

37. Section 124

기본키로 설정할 수 없는 필드는 중복된 값이나 Null 값이 입력된 필드이다.

38. Section 117

- 액세스를 이용하여 테이블을 작성할 때 고려하지 않아도 될 사항은 레코드 수이다.
- 테이블을 작성할 때는 필드 이름, 데이터 형식, 속성 등을

설정할 수 있으며, 속성에는 필드 크기, 유효성 검사, 형식, 입력 마스크 등이 있다.

39. Section 126

그림에서 관계 종류가 일 대 다 관계이므로, <매출> 테이블의 ‘매출거래처’ 필드가 <거래처> 테이블의 ‘거래처번호’를 참조하고 있는 것이다. ‘관련 필드 모두 업데이트’가 설정되어 있으므로 <거래처> 테이블의 ‘거래처번호’가 변경되면 이를 참조하는 <매출> 테이블의 ‘매출거래처’가 자동으로 업데이트된다.

40. Section 130

데이터나 데이터베이스 개체는 Microsoft Excel이나 기타 스프레드시트 프로그램으로 내보낼 수 있으며, 이때 서식이 지정되지 않은 형태로 내보내기 된다.

41. Section 126

관계 설정 시 사용자가 실수로 데이터를 변경하거나 삭제하지 않도록 ‘항상 참조 무결성 유지’를 지정했기 때문에 문제의 메시지가 표시된 것이다. 이럴 경우에는 참조되는 테이블(분류)의 변경 사항을 참조하는 테이블(제품)에 자동으로 반영하도록 ‘관계 편집’ 창의 ‘관련 필드 모두 업데이트’를 선택해야 한다. 이 기능을 이용해야만 정확한 데이터 관리가 가능하다.

3장 정답 및 해설 — 데이터베이스 질의(Query)

1.④ 2.④ 3.④ 4.① 5.④ 6.③ 7.③ 8.③ 9.④ 10.② 11.④ 12.① 13.③ 14.②
15.② 16.④ 17.④ 18.③ 19.① 20.② 21.④ 22.② 23.④ 24.② 25.② 26.③ 27.④ 28.③
29.④ 30.② 31.② 32.② 33.① 34.② 35.④ 36.② 37.② 38.② 39.① 40.① 41.③ 42.①
43.①

1. Section 131

질의 마법사로 생성된 질의도 [SQL 보기] 메뉴를 이용하여 SQL문으로 변경할 수 있다.

2. Section 132

2개 이상의 필드를 나열할 경우에는 필드와 필드를 쉼표(,)로 구분한다.

3. Section 132

Order By절로 오름차순 정렬할 경우 숫자, 한글, 영문(대문자→소문자) 순으로 정렬된다.

※ “Select * from t where 1 = 1”에서 ‘where 1=1’은 ‘1이 1과 같은’이란 조건을 나타내는 것으로 의미 없는 조건이다. 그러므로 “Select * from t”와 “Select * from t where 1 = 1”의 수행 결과는 같다.

4. Section 132

중복 레코드를 제거하기 위한 옵션은 DISTINCT이다.

5. Section 132

SELECT 필드 이름 FROM 테이블 이름 WHERE 조건

6. Section 132

HAVING은 GROUP BY문에 의해 지정된 그룹에 대한 조건을 지정하기 위해 사용한다.

7. Section 133

입력한 자료가 오류(ERROR)인지의 여부를 확인하는 함수는 ISERROR이다. ISNULL 함수는 인수로 지정된 값이 NULL인지 여부를 확인하는 함수이다.

8. Section 134

③과 같이 Or 연산자를 사용하려면 ‘Where 나이 = 18 Or 나이 = 19 Or 나이 = 20 Or 나이 = 21’과 같이 작성해야 한다.

10. Section 133, 134

이름의 첫 글자를 함수로 표현하면 Left(이름,1)이고, 포함되는 값은 IN(‘김’, ‘이’, ‘박’)으로 표현한다.

11. Section 133

질의문을 각 절별로 풀어보면 다음과 같다.

- SELECT 평균, 11(평균)=90, “합격”, “불합격”) AS 합격여부 : ‘평균’과 ‘합격여부’ 필드를 검색하되, 합격여부는 평균이 90 이상이면 ‘합격’을, 그렇지 않으면 ‘불합격’으로 표시한다.

- FROM 성적 : <성적> 테이블을 이용한다.

그러므로 결과에는 평균과 합격여부 필드가 표시되어야 한다.

12. Section 132

지문의 질의문을 각 절별로 설명하면 다음과 같다.

- SELECT 부서, AVG(기본급) AS [부서별기본급평균] : ‘부서’와 ‘부서별기본급평균’을 검색한다. AVG(기본급)을 계산하여 필드명을 ‘부서별기본급평균’으로 표시한다.
- FROM 직원 : <직원> 테이블에서 검색한다.
- GROUP BY 부서 : 부서를 기준으로 그룹을 지정한다.

GROUP BY 절을 생략하면 그룹이 지정되지 않으므로 결과는 다르게 표시된다. 나머지 보기가 옳은 이유는 다음과 같다.

② 평균을 구하는 식 ‘AVG(기본급)’ 대신 ‘SUM(기본급)/COUNT(기본급)’을 사용한 것으로 결과는 같다. 평균은 ‘합계/개수’로 구할 수 있다.

③ ‘부서’와 ‘부서별기본급평균’ 필드만 지정했으므로 필드수는 항상 2개이다.

④ ‘부서’로 그룹을 지정했으므로 레코드는 부서 종류의 개수만큼만 표시된다.

13. Section 133

‘학년’ 필드만 조회하되 DISTINCT문에 의해 중복된 값이 한 번만 표시되므로 결과는 다음과 같다.

학년
1
2

14. Section 134

문제에 제시된 내용을 만족하는 질의문의 코드를 살펴보면 다음과 같다.

```
① SELECT 학번, 과목, 점수
② FROM 성적
③ WHERE 학번 IN (SELECT 학번 FROM 학생 WHERE 학년=1 AND 반=1)
```

- ① 학번, 과목, 점수를 검색한다.
- ② <성적> 테이블에서 검색한다.
- ③ <학생> 테이블에서 학년이 1학년이고 반이 1반인 학생의 학번과 같은 학번을 갖고 있는 <성적> 테이블의 레코드만을 대상으로 검색한다.

15. Section 132

각 절별로 질의문을 작성하면 다음과 같다.

- 반별 학생 수를 검색해야 하므로 'Select 반, Count(학번)' 이다. Count는 비어있지 않은 셀의 개수를 구하는 함수로 Count(학번)은 학번 필드에서 비어 있지 않은 셀의 개수를 구한다.
- <성적> 테이블을 검색하므로 'From 성적' 이다.
- 평균값이 95 이상인 학생을 검색하므로 'Where (국어+영어+수학)/3>=95' 이다.
- 반별로 검색하므로 'Group by 반' 이다.
- 모두 합치면 아래와 같다.

```
Select 반, Count(학번)
From 성적
Where (국어+영어+수학)/3>=95
Group by 반;
```

16. Section 136

일련 번호 필드는 업데이트가 불가능하다.

업데이트 쿼리를 실행할 수 없는 필드

- 계산 결과로 생성된 필드
- 요약 쿼리나 크로스탭 쿼리를 원본으로 사용하는 필드
- 일련 번호 필드
- 통합 쿼리의 필드

17. Section 134

하위 질의에서는 ORDER BY절을 사용할 수 없다.

18. Section 136

추가 쿼리, 즉 삽입 쿼리는 새로운 레코드를 기존 테이블에 추

가하는 쿼리이다. ③번은 테이블 만들기 쿼리에 관한 설명이다.

20. Section 136

실행 쿼리를 실행하면 데이터의 변경 여부를 묻는 대화상자가 표시되는데, 여기서 <예>를 클릭하여 변경된 데이터는 되돌릴 수 없다.

21. Section 133

- ④는 IIF에 대한 설명이다.
- SWITCH(조건1, 실행1, 조건2, 실행2 ...): 조건1이 참이면 실행1을, 조건2가 참이면 실행2를, ... 수행

22. Section 137

문제에 제시된 질의문은 통합 질의로, 여러 테이블의 필드 값을 통합하여 표시하되 같은 레코드는 한번만 표시한다.

23. Section 134

질의문은 각 절을 분리하여 이해하면 쉽다.

- SELECT T1.품번, T2.제조사 : <T1> 테이블의 품번 필드와 <T2> 테이블의 제조사 필드를 검색한다.
- FROM T1, T2 : <T1> 테이블과 <T2> 테이블에서 검색한다.
- WHERE T2.소재지 IN ('서울', '수원') AND T1.품번 = T2.품번 : <T2> 테이블의 소재지가 '서울' 또는 '수원' 이고 <T1> 테이블의 품번과 <T2> 테이블의 품번이 같은 레코드만을 대상으로 검색한다.

24. Section 132

각 절별로 질의문을 작성하면 다음과 같다.

- 월별로 납부금액의 합계를 계산해야 하므로 'select month(납부일) As 월, sum(납부금액)' 이다.
 - month : 날짜에서 월만 표시한다.
 - sum : 납부금액의 합계를 구한다.
- <회비납부> 테이블을 검색하므로 'from 회비납부' 이다.
- 월별로 검색하므로 'group by month(납부일)' 이다.
- 모든 보기에 공통적으로 적용된 조건으로 'where year(납부일) = year(date())' 가 있다. 이것은 올해 회비를 납부한 회원을 대상으로 한다는 의미이다.

```
where year(납부일) = year(date())
```

- year : 날짜에서 년만 표시한다.
- date : 현재 날짜를 표시한다.
- 모두 합치면 다음과 같다.

```
select month(납부일) as 월, sum(납부금액) from 회비납부 where
year(납부일) = year(date()) group by month(납부일)
```

25. Section 133

구매수량의 평균을 구하려면 다음과 같은 방법을 사용할 수 있다.

- ① 평균을 구하는 AVG 함수를 사용한다(=AVG([구매수량])).
- ② 구매수량의 합계를 구한 후 구매 건수로 나눈다.
 - i) =SUM([구매수량])/COUNT([구매수량])
 - ii) =SUM([구매수량])/COUNT(*) : 구매 건수를 구할 때 [구매수량] 대신 모든 레코드 개수를 사용해도 된다.

26. Section 134

문제의 SQL문을 한 줄씩 살펴보면 다음과 같다.

- SELECT CNO → CNO를 검색한다.
- FROM COURSE → <COURSE> 테이블에서 검색한다.
- WHERE CNO LIKE 'K?' → CNO가 'K'로 시작하는 두 글자인 레코드만을 대상으로 검색한다. '?'는 한자리 문자를 대표하는 대표 문자이다.

28. Section 137

- 크로스탭 질의 : 테이블의 특정 필드의 요약값(합계, 개수, 평균 등)을 표시하고 그 값들을 그룹별로, 한 집합은 데이터시트의 왼쪽에, 또 한 집합은 데이터시트의 위쪽에 나열함
- 매개변수 질의 : 레코드 검색 조건이나 필드에 삽입할 값과 같은 정보를 쿼리 실행 시 대화상자로 입력받는 쿼리
- 하위 질의 : 선택 질의(SELECT) 혹은 실행 질의(INSERT, UPDATE, DELETE) 안에 작성하는 SELECT문

29. Section 134

'성명'을 기준으로 그룹(GROUP BY)을 설정한 후 점수에 대한 평균을 계산하면 된다.

그룹 지정 형식

```
SELECT 필드이름
FROM 테이블이름
[WHERE 조건식]
[GROUP BY 필드이름]
[HAVING 그룹 조건식]
```

※ AVG(필드명)은 필드의 평균을, SUM(필드명)은 필드의 합계를 구하는 그룹 함수이다.

30. Section 134

- ① 하위 질의 : SELECT 부서코드 FROM 부서 WHERE 부서이름 = '영업부' : <부서> 테이블에서 부서이름이 '영업부'인 레코드의 부서코드를 추출한다.
- ② SELECT 성명, 근무년수 FROM 사원 WHERE 부서코드 IN (①) : <사원> 테이블의 부서코드 중 추출한 부서코드(①)와 같은 레코드의 성명과 근무년수를 표시한다.

31. Section 137

- ①, ③ 쿼리 실행 결과의 필드는 <성적> 테이블의 '학번', '학과', '점수'로 총 3개의 필드가 표시된다.
- ② UNION은 기본적으로 중복된 레코드는 한 번만 표시하는데, ALL 옵션을 사용하면 중복을 허용하여 표시하므로 총 15개의 레코드가 표시된다.
- ④ 정렬 기준이 생략된 경우에는 오름차순으로 정렬된다.

32. Section 136

테이블로부터 추가할 때는 'INSERT INTO ~ SELECT ~ FROM ~ WHERE' 형식으로 사용하는데, [거래처] 테이블에 존재하지 않는 데이터만을 대상으로 해야 하므로 하위 질의를 not in으로 연결하여 'Insert into 거래처 select * from 주소록 where 거래처번호 not in (select 거래처번호 from 거래처)'으로 작성하면 된다.

34. Section 135

INNER JOIN 일반 형식

```
SELECT 필드이름
FROM 테이블이름1 INNER JOIN 테이블이름2
ON 테이블이름1.필드이름 = 테이블이름2.필드이름
```

35. Section 135

보통 연결될 두 테이블 간에 관계가 설정되어 있어야 하지만, 관계가 설정되지 않아도 조인을 수행할 수는 있다.

36. Section 134

정렬 질의문 형식

```
SELECT 필드이름
FROM 테이블이름
[WHERE 조건식]
[ORDER BY 필드이름, 정렬방식]
```

- ORDER BY문 : 특정 필드를 기준으로 레코드를 정렬하여 검색할 때 사용함

- 정렬 방식 : 'ASC' 와 'DESC' 가 있으며, 'ASC' 는 오름차순, 'DESC' 는 내림차순을 의미함

37. Section 134

- < > 0 Or < 4 : 0과 같지 않거나 4보다 작은 값을 입력 받는다.
- Like < > 0 And < 4 : 사용 형식이 잘못되었다. 'LIKE "*용산구"' 형식으로 사용해야 한다.
- Between < > 0 And < 4 : 사용 형식이 잘못되었다. 'Between 1 and 3' 형식으로 사용해야 한다.

38. Section 132

지문의 질의문을 절별로 살펴보면 다음과 같다.

- SELECT DISTINCT COUNT(성명) : '성명' 필드가 NULL이 아닌 레코드의 개수를 세어 표시한다. 여기서 DISTINCT는 결과에 대한 중복 값을 제거하는 것으로서 의미 없이 사용되었다.
- FROM 회원 : <회원> 테이블을 검색한다.

그러므로 결과는 성명이 NULL이 아닌 레코드의 개수 3이 된다.

39. Section 133

각 질의문의 결과는 다음과 같다.

- Update 회원 Set 회원번호 = Right("00"& 회원번호,3)
Right("00"& 회원번호,3)
①
②
① 회원번호 앞에 "00"을 연결한다(1 → 001, 78 → 0078)
② Right(①, 3) : ① 문자열의 오른쪽에서 3자리를 추출한다(001, 0078).
- Update 회원 Set 회원번호 = Left("00"& 회원번호,3)
Left("00"& 회원번호,3)
①
②
① 회원번호 앞에 "00"을 연결한다(1 → 001, 78 → 0078)
② Left(①, 3) : ① 문자열의 왼쪽에서 3자리를 추출한다(001, 007). 처음에 입력된 회원번호가 한 자리인 경우(1)에는 정상적으로 변경되지만 78과 같이 두 자리인 경우에는 "0078" 중 왼쪽에서 3자리를 추출하므로 결과는 "007"로 변경된다. 그러므로 올바른 수식이 아니다.
- Update 회원 Set 회원번호 = Right(회원번호& "00",3)
Right(회원번호& "00",3)
①
②
① 회원번호 뒤에 "00"을 연결한다(1 → 100, 78 → 7800).
② Right(①, 3) : ① 문자열의 오른쪽에서 3자리를 추출한

다(100, 800).

- Update 회원 Set 회원번호 = Left(회원번호& "00",3)
Left(회원번호& "00",3)

- ① 회원번호 뒤에 "00"을 연결한다(1 → 100, 78 → 7800).
- ② Left(①, 3) : ① 문자열의 왼쪽에서 3자리를 추출한다(100, 780).

40. Section 137

- 크로스탭 쿼리는 테이블의 특정 필드를 그룹별로 요약(합계, 개수, 평균 등)하고 요약된 값들에 대한 제목을 데이터 시트의 왼쪽과 오른쪽에 표시한다.
- 쿼리를 실행할 때 레코드 검색 조건이나 필드에 삽입할 값과 같은 정보를 입력받아 실행하는 것은 매개 변수 쿼리이다.

41. Section 132

Where절은 [도서] 테이블과 [주문] 테이블에 동일하게 들어 있는 도서번호에 해당하는 레코드만 추출하기 위한 것으로, Where절이 없으면 [도서] 테이블과 [주문] 테이블의 모든 레코드들이 연결되는 교차곱(Cartesian Product) 연산이 수행되며, 교차곱 수행의 결과에 Group By절이 적용되어 도서명을 기준으로 그룹화 되어 표시된다.

42. Section 134

'서울' 이나 '인천' 이 아닌, 즉 '서울' 과 '인천' 을 포함하지 않는 것을 검색하면 되므로 NOT IN ('서울', '인천')으로 작성하면 된다. ③의 조건인 'Where(거주지 < > '서울') Or (거주지 < > '인천')' 는 거주지가 '서울' 이 아니거나 '인천' 이 아니라는 조건이므로 조건으로서 의미가 없다.

43. Section 135

관계가 설정된 두 테이블에서 조인된 필드가 일치하는 행만 질의에 포함되는 조인은 내부 조인(Inner Join)이다.

4장 정답 및 해설 — 품과 컨트롤

1.③ 2.④ 3.④ 4.③ 5.② 6.④ 7.② 8.③ 9.④ 10.④ 11.④ 12.③ 13.② 14.①
 15.① 16.④ 17.② 18.③ 19.① 20.④ 21.② 22.③ 23.④ 24.③ 25.④ 26.① 27.④ 28.③
 29.③ 30.④ 31.② 32.② 33.④ 34.④ 35.③ 36.② 37.② 38.① 39.③ 40.④ 41.④ 42.②
 43.④ 44.② 45.② 46.② 47.③ 48.③ 49.② 50.③

1. Section 139

특정 레코드만을 품에 표시하려면 원하는 레코드만 표시되는 쿼리를 만들어 품의 원본 데이터로 지정하면 된다.

2. Section 141

레코드 원본이 아닌 테이블의 필드 값을 컨트롤에 표시하려면 컨트롤의 컨트롤 원본 입력란에서 식 작성기 단추를 클릭한 후 표시할 테이블의 필드를 설정하면 된다.

3. Section 139

[그룹화 및 정렬] 기능은 보고서 작성 시 사용할 수 있는 기능이다.

4. Section 141

읽기 전용이란 기존 내용에 변화를 줄 수 있는 추가, 삭제, 수정(편집) 작업을 할 수 없도록 잠가두는 것으로 추가 가능, 삭제 가능, 편집 가능 속성을 '아니오'로, 잠금 속성을 '예'로 설정해야 한다.

5. Section 139

- ②는 탭 형식의 품을 디자인 보기 형식으로 전환한 것이다.
- 맞춤 모양은 다음 그림과 같다.



6. Section 140

품이 이미 실행된 상태에서 각 형식으로 전환하려면 [품 디자인 도구] → [디자인] → [보기] 그룹을 이용하여 전환해야 한다.

7. Section 140

데이터시트 보기는 원본 데이터를 행과 열 서식으로 표시하

는 형식으로 컨트롤의 속성은 변경할 수 없다. 컨트롤의 속성은 디자인 보기 형태에서 변경이 가능하다.

8. Section 140

- 품의 디자인 보기 상태는 품을 작성하고 수정하는 형식으로, 필드 추가·변경, 컨트롤 추가·변경, 각 개체의 속성 지정 등을 수행할 수 있다.
- 데이터를 추가하거나 삭제, 변경하려면 품 보기나 데이터 시트 보기에서 수행해야 한다.

9. Section 140

[품 디자인 도구] → [디자인] → [보기]에서는 '전체 화면'을 제공하지 않는다.

10. Section 144

콤보 상자나 목록 상자의 행 원본에서는 테이블이나 쿼리, 사용자가 직접 입력한 값, 필드 목록을 사용할 수 있다.

11. Section 141

제목 표시줄을 더블클릭하면 속성 시트 창이 표시되지 않고 품 화면이 최대화 된다.

12. Section 141

③은 '기본 보기' 속성에 대한 설명이다.

13. Section 139

세부 정보 필드에 여러 개의 필드를 삽입하여 분석할 수 있다.

14. Section 141

품의 테두리 스타일 중에서 제목 표시줄, 닫기 단추, 조절 매뉴만 표시하는 형태는 '대화 상자'이다.

15. Section 142

계산 컨트롤은 원본 데이터로 필드를 사용하지 않고 식을 사용하는 컨트롤이지만, 식도 '컨트롤 원본'에 작성한다.

16. Section 142

명령 단추 컨트롤의 이벤트를 설정하여 매크로 함수뿐만 아니라 이벤트 프로시저도 수행되게 할 수 있다.

17. Section 142

- 텍스트 상자 : 폼이나 보고서의 원본으로 사용되는 데이터나 계산 결과를 표시하는 컨트롤
- 명령 단추 : 레코드를 찾거나 레코드 인쇄 등의 특정 기능을 실행할 때 사용하는 컨트롤
- 콤보 상자 : 텍스트 상자와 목록 상자가 결합된 형태로, 좁은 공간에서 많은 자료를 표현할 때 유용하게 사용되는 컨트롤

18. Section 142

- 콤보 상자 : 텍스트 상자와 목록 상자가 결합된 형태로, 좁은 공간에서 유용하게 사용되는 컨트롤이며, 데이터를 목록에서 선택하거나 직접 입력할 수 있음
- 목록 상자 : 콤보 상자와 비슷한 컨트롤로, 데이터를 목록에서만 선택할 수 있는 컨트롤임

19. Section 142

일반적으로 텍스트 상자는 컨트롤 원본 속성에 설정한 필드의 값이나 식의 결과를 표시한다.

20. Section 144

콤보 상자와 목록 상자의 속성을 살펴보면 대부분이 동일하다. 하지만 ‘행 수’, ‘목록 값만 허용’ 등의 속성은 콤보 상자에만 있는데, 이는 콤보 상자에서는 목록 이외의 값을 입력할 수 있고 목록 상자에서는 목록 이외의 값을 입력할 수 없음을 나타낸다.

21. Section 142

- 레이블 : 제목이나 캡션, 설명 등과 같은 텍스트를 표시하는 컨트롤로, 언바운드 컨트롤
- 확인란 : 여러 개의 값 중 하나 이상을 선택할 수 있는 컨트롤
- 콤보 상자 : 텍스트 상자와 목록 상자가 결합된 형태로, 좁은 공간에 유용하게 사용됨

22. Section 145

하위 폼은 단일 폼, 연속 폼, 데이터시트, 피벗 테이블, 피벗 차트 보기 형식 등으로 표시할 수 있다.

23. Section 145

다음과 같은 조건을 만족할 때 기본 폼과 하위 폼이 자동으로

연결된다.

- 기본 폼과 하위 폼에서 사용되는 테이블/쿼리 간에 관계가 설정된 경우
- 기본 폼은 기본키를 가진 테이블/쿼리를 사용하고, 하위 폼은 기본키 필드와 같은 이름을 갖고, 같거나 호환되는 데이터 형식, 필드 크기를 가진 필드가 포함된 테이블/쿼리를 사용할 경우

24. Section 145

- ① ‘직접 지정’은 관계 설정 여부에 관계 없이 사용자가 직접 기본 폼과 하위 폼을 연결할 필드를 선택할 때 사용한다.
- ② 연결 필드의 지정 및 변경은 [데이터] 탭의 ‘하위 필드 연결’이나 ‘기본 필드 연결’ 어느 곳에서나 가능하다.
- ④ 하위 폼 연결기 창에서는 기본 폼과 하위 폼의 연결 필드를 한꺼번에 지정할 수 있다.

26. Section 148

도메인 함수에서 사용되는 인수는 각각을 큰따옴표(“”)로 묶어줘야 하며 문자열을 연결할 때에는 &를 사용한다.

27. Section 146

- ①, ②, ③에는 **Shift**가 사용되고, ④에는 **Ctrl**이 사용된다.

28. Section 146

세개의 컨트롤을 선택한 뒤 [폼디자인 도구] → [정렬] → [위치] → [가로 간격 같음]을 선택하면 가장 왼쪽과 오른쪽 컨트롤의 위치를 기준으로 하여 그 사이에 있는 컨트롤의 위치가 변경되어 간격이 동일하게 조정된다.

29. Section 147

- [형식] 탭 : 하이퍼링크, 중복 내용 숨기기
- [데이터] 탭 : 유효성 검사 규칙, 기본값
- [기타] 탭 : IME 모드, 데이터시트 캡션, 상태 표시줄 텍스트, 탭 인덱스

30. Section 147

행 원본 유형은 콤보 상자와 목록 상자에서, 확장 기능은 텍스트 상자에서, 특수 효과는 대부분의 컨트롤에서 설정할 수 있다.

31. Section 147

‘탭 순서’ 대화상자에서 <자동 순서> 단추를 클릭하면 폼이나 보고서에 삽입되어 있는 컨트롤의 위치를 기준으로 탭 순서를 위쪽에서 아래쪽, 왼쪽에서 오른쪽 순으로 자동 설정된다.

32. Section 147

컨트롤 원본은 연결할 데이터 설정, 바운드 열은 콤보 상자, 목록 상자 컨트롤에 저장할 열 설정, 여러 항목 선택은 목록 상자에서 여러 항목의 선택 여부와 방법을 설정하는 속성이다.

33. Section 148

연속 폼에서 폼의 아랫 부분에 오늘 날짜와 시간을 표시하려면, 폼 바닥글에 텍스트 상자 컨트롤을 만들어 컨트롤 원본에 오늘 날짜와 시간을 표시하는 now() 함수를 입력하면 된다.

34. Section 148

‘페이지 번호’ 대화상자에서는 페이지 번호를 페이지 머리글이나 페이지 바닥글에 표시되도록 설정할 수 있다.

35. Section 141

(3)번 부분을 더블클릭하면 (1)번을 더블클릭한 것과 같이 폼의 속성 시트 창이 표시된다. 폼 바닥글의 배경색을 변경하려면 폼 바닥글 영역을 더블클릭해야 한다.

36. Section 142

- 옵션 그룹 : 여러 개의 값 중 하나를 선택할 수 있는 컨트롤
- 탭 컨트롤 : 탭 형식의 대화상자를 작성하는 컨트롤
- 목록 상자 : 콤보 상자와 비슷한 컨트롤로, 목록의 데이터만 사용할 수 있는 형태의 컨트롤

38. Section 145

하위 폼은 일 대 다 관계에 있는 테이블이나 쿼리를 효과적으로 표시하는 컨트롤로 ‘일’에는 기본 폼, ‘다’에는 하위 폼을 배치한다.

39. Section 145

하위 폼은 단일 폼, 연속 폼, 데이터시트 등으로 표시가 가능하다.

40. Section 145

식에 필드의 이름을 사용할 때는 필드 이름을 대괄호 ([])로 묶는다.

42. Section 142

- 콤보 상자 : 텍스트 상자와 목록 상자가 결합한 형태로, 데이터를 목록에서 선택하거나 직접 입력할 수 있음
- 목록 상자 : 콤보 상자와 비슷한 컨트롤로, 목록의 데이터만 사용할 수 있는 형태의 컨트롤

43. Section 145

‘기본 필드 연결’ 속성에는 기본 폼과 하위 폼을 연결할 때 사용하는 기본 폼의 연결 필드를 지정해야 한다. 표시할 폼은 하위 폼/하위 보고서 마법사로 하위 폼/보고서를 만들 때 지정한다.

44. Section 146

‘중복 내용 숨기기’ 속성은 레이블 컨트롤이 아니라 텍스트 상자 컨트롤의 속성이다.

45. Section 147

컨트롤 원본 속성을 비워두면 컨트롤에 아무것도 표시되지 않으며 수정도 할 수 없다.

46. Section 151

폼 실행 시 [Alt]를 이용하여 명령 버튼을 클릭하려면 명령 버튼의 캡션 속성에 ‘&’와 원하는 키를 입력해야 한다.

47. Section 141

〈A화면〉에서 레이블이 레코드마다 매번 표시되는 이유는 레이블이 본문 영역에 있기 때문이다. 본문 영역에 있는 레이블을 폼 머리글로 이동하면 〈B화면〉과 같이 상단에 한 번만 표시된다. 나머지 보기의 내용대로 지정했을 경우

- ① ‘기본 보기’ 속성을 연속 폼으로 선택하였을 경우 : 문제의 〈A화면〉이 ‘기본 보기’ 속성을 연속 폼으로 선택한 것임

② '기본 보기' 속성을 단일 폼으로 선택하였을 경우

④ 본문의 모든 텍스트 상자를 폼 머리글로 옮긴 경우

48. Section 141

텍스트 상자 컨트롤의 이름을 변경하려면 텍스트 상자 컨트롤의 바로 가기 메뉴에서 [속성]을 선택한 후 '기타' 탭의 '이름' 속성에 원하는 이름을 입력하면 된다.

49. Section 139

폼 분할

- 하나의 원본 데이터를 이용하여 위쪽 구역에는 데이터시트를 표시하는 폼을 만들고, 아래쪽 구역에는 데이터시트에서 선택한 레코드에 대한 정보를 입력할 수 있는 폼을 작성한다.
- 두 형식은 하나의 원본 데이터를 사용하므로 서로 연결되어 있어 항상 동기화 된다.
- 디자인 보기 상태로 열면 컬럼 형식의 형태로 컨트롤들이 표시된다.

50. Section 139

모달 대화상자 도구를 사용하여 폼을 만들면 <확인>과 <취소> 버튼이 자동으로 생성되는데, 자동으로 생성된 <확인>과 <취소> 버튼에는 기본적으로 현재 개체를 닫는 Close 매크로 함수가 설정되어 있다. 그러므로 모달 대화상자가 실행된 상태에서 <확인>이나 <취소> 버튼을 클릭하면 대화상자가 닫힌다.

5장 정답 및 해설 — 보고서(Report) 작성

- 1.④ 2.④ 3.③ 4.④ 5.③ 6.④ 7.① 8.③ 9.④ 10.① 11.③ 12.① 13.④ 14.③
15.① 16.① 17.③ 18.③ 19.② 20.② 21.④ 22.③ 23.① 24.② 25.④ 26.① 27.④ 28.③
29.④ 30.② 31.② 32.④ 33.④ 34.④ 35.② 36.② 37.① 38.③ 39.④ 40.④ 41.③ 42.②
43.③

1. Section 149, 150, 152

- ① '파일로 인쇄'를 선택하면 보고서를 저장하는 파일의 확장자는 '*.prn'이다.
- ② 레이블 보고서는 [만들기] → [보고서] → [보고서 디자인]을 클릭 한 후 사용자가 직접 만들 수도 있다.
- ③ 보고서에서도 컨트롤 원본을 이용하여 컨트롤을 특정 필드에 바운드시킬 수 있다.

2. Section 149

보고서 바닥글은 디자인 보기 상태에서는 가장 마지막 구역에 표시되고, 인쇄 미리 보기 상태에서는 마지막 페이지의 페이지 바닥글 위쪽에 한 번만 표시된다.

3. Section 149

보고서는 이미 만들어진 테이블이나 질의 등의 데이터를 요약하거나 그룹화하여 종이에 출력하기 위한 개체이다.

4. Section 149

제품명이 첫 번째만 표시되고 그 다음에 표시되지 않는 것은 '중복 내용 숨기기' 속성을 '예'로 지정하였기 때문이다. '제품명' 컨트롤을 그룹 머리글 영역에 작성하였다면 다음 그림과 같이 제품명과 납품일자, 거래처명 등이 같은 행에 표시되지 않는다.

제품명	납품일자	거래처명	납품단가	납품수량	납품금액
와이어	2003-11-14	강진전기	40	43	1,720
	2003-10-11	서울전기	40	34	1,360
	2003-10-09	여수전기	40	30	1,200

- ① ‘제품별 납품 현황’ 이 전체 7 페이지 중 5번째 페이지에도 표시된 것으로 보아 페이지 머리글에 작성하였음을 알 수 있다.
- ② 제품명, 납품일자, 거래처명 등의 레이블과 두 개의 선이 그룹이 바뀔 때마다 해당 그룹의 위에 표시되는 것으로 보아 그룹 머리글에 작성한 후 ‘반복 실행 구역’ 속성을 ‘예’로 설정하였음을 알 수 있다.
- ③ 본문 영역에 표시된 레코드 중 특정 레코드에만 바탕색이 지정된 것으로 보아 조건부 서식이 지정되었음을 알 수 있고, 바탕색이 지정된 레코드의 납품수량이 모두 100보다 크고, 본문에서 제품명을 제외한 모든 컨트롤에 서식이 지정되어 있으므로 본문 영역에서 제품명을 제외한 모든 컨트롤을 선택한 후 ‘조건부 서식’ 대화상자에서 조건을 ‘[납품수량]>100’으로 지정하였음을 알 수 있다.

5. Section 149

페이지 바닥글은 각 페이지 하단에 날짜나 페이지 번호 등을 삽입할 때 사용하는 영역으로, SUM 함수를 사용할 수 없다.

6. Section 149

보고서에서는 데이터 입력, 추가, 삭제 등의 작업을 할 수 없다.

7. Section 149

실제 데이터가 반복적으로 표시되는 부분은 본문이다.

8. Section 155

‘납품수량’ 이 100보다 크거나 같으면 나머지 필드에도, 즉 조건이 만족하는 레코드 전체에 대해서 서식을 표시하려면 본문의 모든 컨트롤을 선택한 후 [보고서 디자인 도구] → [디자인] → [글꼴] → [조건부 서식]을 선택한 다음 ‘조건부 서식’ 대화상자에서 보기 ③번과 같이 지정하면 된다.

9. Section 149

- 보고서 머리글 영역 : 보고서의 첫 페이지 상단에 한 번 표시됨(예를 들어, 10쪽짜리 출력물이라면 첫 페이지의 맨 윗부분)
- 본문 영역 : 실제 데이터가 반복적으로 표시되는 부분
- 그룹 머리글 영역 : 그룹이 지정될 경우 그룹의 상단에 반복적으로 표시됨

- 페이지 머리글 영역 : 각 페이지마다 상단에 표시되며, 주로 열 제목 등을 삽입함

10. Section 149

매 페이지마다 표시될 항목은 보고서 머리글이 아니라 페이지 머리글에 설정해야 한다.

11. Section 149

그룹 바닥글은 그룹의 하단에 반복적으로 표시되는 것으로 보고서 전체에 적용되는 정보를 표시하는 것이 아니라 그룹별 요약 정보 등을 표시할 때 사용한다.

12. Section 150

- 제목 표시는 레이블 컨트롤을 이용한다.
- ②는 옵션 그룹, ③은 텍스트 상자, ④는 하위 폼/하위 보고서 컨트롤이다.

13. Section 151

보고서의 인쇄 미리 보기 상태에서는 마우스로 드래그하여 페이지 여백을 재설정할 수 없다.

14. Section 151

‘편집 가능’ 속성은 폼의 속성이다. 보고서에서는 데이터 입력, 추가, 삭제 등의 작업을 할 수 없다.

15. Section 152

- 문자열을 연결할 때는 & 연산자를 사용한다.
- 문자열 연결 시 단순 문자열은 “ ”로 묶고, 필드나 명령어는 묶지 않는다.
- [Pages]는 전체 페이지 수, [Page]는 현재 페이지 번호를 표시하다.

17. Section 155

컨트롤 원본을 ‘=1’로 설정하고 누적 합계 속성을 ‘모두’로 설정하면 그룹에 관계없이 보고서의 끝까지 값이 누적된다. 그룹별로 순번(일련번호)을 표시하려면 누적 합계 속성을 ‘그룹’으로 설정해야 한다.

18. Section 149

- 보고서의 첫 번째 페이지에만 나타나는 내용은 보고서 머리글에서 지정할 수 있다.
- 그룹 머리글 : 그룹의 상단에 반복적으로 표시됨

19. Section 155

보고서 바닥글에 입력했으므로 전체 레코드의 개수를 계산한다. 요약정보표시 함수는 입력하는 위치에 따라 그룹을 대상으로 할지 전체를 대상으로 할지가 결정된다.

20. Section 154

크로스탭 보고서는 크로스탭 쿼리를 레코드 원본으로 하여 작성한다.

21. Section 155

‘조건부 서식’ 대화상자의 <삭제> 단추를 눌러 적용된 서식을 해제할 수 있다.

22. Section 155

- 필드의 총 개수를 구하면 되므로, ‘=COUNT(*)’를 사용한다.
- ①은 평균, ②는 최대값, ④는 날짜와 시간을 계산한다.

23. Section 155

③은 ‘보통 날짜’, ④는 ‘간단한 날짜’ 형식이다.

24. Section 153

보고서에서 그룹을 만들려면 머리글 구역 표시나 바닥글 구역 표시 중 하나 이상을 선택하면 된다.

25. Section 149

- 보고서 머리글의 내용은 보고서의 첫 페이지 상단에 한 번 인쇄한다.
- 보고서 바닥글의 내용은 보고서의 마지막 페이지 하단에 한 번 인쇄한다.
- 보고서에서는 필드나 식을 기준으로 10개까지 정렬하거나 그룹화할 수 있다.

26. Section 155

- Format() : 표시 형식 지정
- Date() : 현재 날짜 표시
- mmm : 월을 Jan ~ Dec로 표시

27. Section 149

보고서 바닥글은 디자인 보기 상태에서는 가장 마지막 구역에 표시되지만, 인쇄하거나 미리 보는 경우에는 마지막 페이지의 페이지 바닥글 위쪽에 한 번만 표시된다.

28. Section 153

텍스트 형식의 필드에서는 전체 필드 값이 일치할 경우 그룹화하는 ‘전체 값’, 첫 문자가 같을 경우 그룹화하는 ‘첫 문자’, 시작하는 두 문자가 같을 경우 그룹화하는 ‘처음 두 문자’, 사용자가 지정한 문자 수 만큼만 같으면 그룹화하는 ‘사용자 지정 문자’ 중 하나를 선택하여 그룹을 지정할 수 있다.

29. Section 153

보고서에서는 필드나 식을 기준으로 10개까지 그룹 수준을 만들 수 있다.

30. Section 155

- AVG(인수) : 인수의 평균 계산
- ABS(인수) : 인수의 절대값 계산
- COUNT(인수) : 인수의 개수 계산
- MAX(인수) : 인수 중 최대값 계산

31. Section 155

도서 목록이라는 제목 등이 표시된 음영 부분은 현재 페이지 인 4 페이지에도 표시되어 있으므로 그룹 머리글이 아닌 페이지 머리글에 작성된 것이다. 페이지 머리글에는 ‘반복 실행 구역’ 속성이 없다.

다른 보기가 옳은 이유는 다음과 같다.

- 출판사명을 기준으로 구분되어 있고, 각 그룹 시작 부분에 점선이 표시되어 있으므로 ‘출판사명’ 필드로 그룹이 설정되어 있다는 것과 그룹 머리글 영역이 설정되어 있다는 것을 알 수 있다.
- ‘출판사명’과 ‘저자명’이 오름차순 정렬되어 있고 상단에 한번만 표시되어 있는 것으로 보아 ‘중복 내용 숨기기’ 속성이 ‘예’로 설정되어 있음을 알 수 있다.
- ‘순번’ 필드는 그룹에서 1부터 시작하여 1씩 증가하므로 컨트롤 원본에 ‘=1’을 설정하고, ‘누적 합계’ 속성을 ‘그룹’으로 설정했음을 알 수 있다.

32. Section 154

- 차트 보고서 : 데이터를 막대나 선, 도형, 그림 등을 사용하여 시각적으로 표현한 차트를 이용하여 보고서를 작성한 것
- 크로스탭 보고서 : 보고서를 가로, 세로 방향으로 모두 그룹화하고, 그룹화한 데이터에 대해 합, 개수, 평균 등의 계산을 수행한 것
- 레이블 보고서 : 많은 양의 우편물을 발송할 때 쉽고 간단하게 주소를 출력할 수 있도록 만든 것

33. Section 149

- 보고서 바닥글 : 보고서의 맨 마지막 페이지에 한 번 표시되는 부분으로 보고서 총계, 안내 문구 등을 삽입함
- 페이지 바닥글 : 각 페이지의 하단에 표시되며, 주로 날짜나 페이지 번호를 삽입함

34. Section 149

어떤 컨트롤을 사용하든지에 상관없이 보고서에서는 데이터 입력 및 편집이 불가능하다.

35. Section 149

필드의 이름을 표시하는 레이블인 '도서 분류', '도서명', '저자', '대출일', '연체료'와 선 컨트롤이 5페이지 중 현재 페이지인 2페이지의 맨 위에 한 번만 표시되어있으므로 '도서분류' 머리글이 아닌 페이지 머리글에 지정되었음을 알 수 있다. 나머지 보기가 옳은 이유는 다음과 같다.

- ① '도서분류'를 기준으로 데이터가 모여 있으므로 그룹 기준이 '도서분류'임을 알 수 있다.
- ③ 다른 그룹이 시작되기 전 점선이 표시되어 있으므로 점선이 '도서분류' 바닥글에 작성되었음을 알 수 있다.
- ④ 각 그룹별 도서의 이름이 상단에 한 번만 표시되어 있으므로 '도서분류' 컨트롤의 '중복 내용 숨기기' 속성이 '예'로 설정되었음을 알 수 있다.

36. Section 150

보고서는 테이블, 쿼리, SQL 문 등을 이용한 데이터 덩어리를 레코드 원본으로 하여 작성할 수 있다.

37. Section 155

- ② "1/5 페이지"의 형태로 표시하려면 "[Page] & "/" & [Pages] & "페이지"로 입력해야 한다.

- ③ =Format([Pages], "000") 식에서 'Pages'는 전체 페이지를 나타내므로 결과는 '005'가 된다.

- ④ 보고서의 모든 페이지 아랫부분에 페이지 번호를 출력하고자 하면 보고서 바닥글이 아니라 페이지 바닥글을 이용해야 한다.

38. Section 149

그룹 머리글에는 그룹 상단에 반복적으로 표시될 이름이나 요약 정보 등을 표시한다. 전체 보고서에 대한 요약 값은 보고서 머리글이나 보고서 바닥글 영역에 표시해야 한다.

39. Section 153

그룹 머리글은 기본적으로 해당 그룹의 시작 부분에 표시되지만, 새로운 페이지가 시작될 때도 그룹 머리글을 표시되게 하려면 '반복 실행 구역' 속성을 '예'로 지정해야 한다.

40. Section 149

보고서에서는 데이터의 입력 및 수정이 불가능하다.

41. Section 153

출판사별 도서수는 그룹 바닥글에 표시되어 있다.

42. Section 153

- ① 보고서에서 필드나 식을 기준으로 10개까지 정렬할 수 있다.
- ③ 정렬할 필드나 식을 선택하면 기본적으로 오름차순으로 설정된다.
- ④ 그룹별로 머리글과 바닥글을 지정할 수 있다.

6장 정답 및 해설 — 데이터베이스 프로그래밍

- 1.② 2.③ 3.④ 4.② 5.③ 6.③ 7.① 8.③ 9.③ 10.② 11.④ 12.③ 13.④ 14.②
15.③ 16.③ 17.② 18.② 19.③ 20.② 21.③ 22.④ 23.④ 24.② 25.④ 26.④ 27.③ 28.③

1. Section 157

'AutoExec'라는 이름의 자동 매크로는 데이터베이스 파일이 열릴 때 자동으로 수행되는 매크로이다.

2. Section 157

기능키를 지정할 경우에는 해당 키를 { }로 묶어준다.

3. Section 157

자동 매크로 실행에 의해 액세스 파일을 열 때 수행되는 매크로는 생성할 수 있어도, 일반 응용 프로그램이 실행될 때 수행되는 매크로는 생성할 수 없다.

5. Section 156

그룹으로 작성된 매크로를 실행할 경우에는 '그룹매크로이름.매크로이름' 과 같이 마침표(.)로 구분한다.

7. Section 158

- FindRecord : 특정한 조건에 맞는 첫 번째 레코드를 검색함
- OpenQuery : 작성된 쿼리를 호출하여 실행함(데이터시트, 디자인, 인쇄 미리 보기, 피벗 테이블, 피벗 차트)
- SendObject : 데이터베이스 개체를 전자우편 메시지에 첨부하여 전송함

8. Section 158, 159

- Maximize 함수에 의해 화면에 나타난 폼은 화면 크기만큼 최대화된다.
- Minimize : 현재 활성화된 창을 작업 표시줄에 아이콘 모양으로 최소화하는 함수

9. Section 158

- StopMacro : 현재 실행중인 매크로를 중지함
- ApplyFilter : 테이블이나 쿼리로부터 레코드를 필터링함
- TransferDatabase : 다른 데이터베이스 파일과의 내보내기, 가져오기, 연결 등을 지원함

10. Section 158

- 그림의 인수는 총점 필드에 국어와 영어 필드의 합을 할당하는 것으로, SetValue 함수의 인수들이다.
- OpenReport : 작성된 보고서를 호출하여 실행함
- ApplyFilter : 테이블이나 쿼리로부터 레코드를 필터링함
- RunMacro : 매크로를 실행함

11. Section 159

- Maximize : 현재 활성화된 창을 최대화하는 함수
- Minimize : 현재 활성화된 창을 작업 표시줄에 아이콘 모양으로 최소화하는 함수
- MoveSize : 현재 활성화된 창의 크기를 변경하는 함수

12. Section 159

SetValue 함수는 폼이나 보고서의 필드, 컨트롤의 속성 등에 어떤 값을 할당할 때 사용한다.

13. Section 158

매크로 함수 OutputTo에서 개체 유형이 테이블인 경우에는 서식 있는 텍스트(*.rtf), 텍스트 파일(*.txt), 통합 문서 파일(*.xls, *.xlsx), *.html, *.pdf, *.xps 등의 형식으로 내보낼 수 있다. Snapshot(*.snp) 형식으로 내보낼 수 있는 개체는 보고서이다.

14. Section 158

조건절 생략 기호(...)

조건절 생략 기호를 조건절에 입력하면 윗 행의 조건이 참인 경우 함께 실행될 매크로 함수로 간주된다.

15. Section 158

CancelEvent 매크로 함수에는 인수가 없다.

16. Section 160

- AfterUpdate : 컨트롤이나 레코드의 데이터가 업데이트된 후에 발생
- Current : 포커스가 임의의 레코드로 이동되어 그 레코드가 현재 레코드가 되거나 폼이 새로 고쳐지거나 다시 질의될 때 발생
- DblClick : 컨트롤이나 레이블을 마우스 왼쪽 단추로 두 번 클릭할 때 발생

17. Section 160

각 코드에 대한 설명은 다음과 같다.

```
Private Sub Form_Load( )  
  ① Dim exstring  
  ② exstring = "Access 2000"  
  ③ Mid(exstring, 3, 5) = "Excel"  
  ④ MsgBox exstring  
End Sub
```

- ① 'exstring' 변수를 사용하기 위해 선언한다.
- ② exstring 변수에 "Access 2000" 텍스트를 저장한다.
- ③ exstring 변수에 있는 텍스트 "Access 2000"의 세 번째 문자부터 5글자(cess) 대신 "Excel"을 저장한다 (AcExcel2000).
※ MID(텍스트, 시작 위치, 개수) : 텍스트의 시작 위치부터 지정한 개수만큼 표시한다.
- ④ exstring 변수에 있는 내용을 메시지 박스(MsgBox)로 표시한다.

18. Section 160

KeyUp

컨트롤이나 폼이 포커스를 가지고 있는 동안 누르고 있던 키를 놓을 때 발생

19. Section 159

Application.Run은 Access나 사용자 정의 Function 또는 Sub 프로시저를 수행한다.

20. Section 161

주민번호 필드의 8번째 자리 값이 1인 경우 SetValue, Beep, MsgBox 함수가 수행된다.

21. Section 161

- Requery : 폼이나 컨트롤의 데이터 원본을 다시 읽어 데이터를 업데이트함
- Refresh : 데이터 원본으로 사용하는 레코드를 즉시 업데이트함
- Undo : 값이 변경된 컨트롤이나 폼을 원래대로 되돌림

22. Section 161

④의 'intTotal = Me.Pages'는 전체 페이지 수를 'intTotal' 변수에 치환하는 것이다.

23. Section 161

RunSQL의 질의에서 두 개의 조건이 and로 연결되어 있으므로, 두 조건이 모두 만족하는 레코드를 삭제한다.

24. Section 160

- GotFocus : 컨트롤이나 폼이 포커스를 받을 때 발생함
- Enter : 폼의 다른 컨트롤에서 특정 컨트롤로 포커스가 실제로 옮겨지기 전이나 폼을 열 때 발생함
- Activate : 폼이나 보고서가 활성화 될 때 발생함

25. Section 162

- Open : 데이터 원본으로 연결된 개체를 오픈
- Close : 열려 있는 개체와 관련된 종속 개체를 모두 닫음
- State : Connection 개체의 속성 중 하나로, 사용할 수 있는 개체의 상태(열림, 닫힘)를 표시함

26. Section 161

- Open() : 폼을 열어 레코드를 처음으로 표시하기 전이나, 보고서를 열어 인쇄하기 전에 발생함
- Load() : 폼을 열어 레코드들이 표시될 때 발생함

- Initialize() : 개체를 로드한 후 그 개체가 보이기 전에 발생함

27. Section 161

Close는 특정 개체를 닫는 명령이고, Quit는 액세스 프로그램을 종료하는 명령이다.

28. Section 158

활성화된 폼에서 커서를 특정한 컨트롤(Control)로 이동(GoTo)시키는 매크로 함수는 GoToControl이다.