



시험에 나오는것만 공부한다!

시나공시리즈

2022년 2회 실기 시험 대비 정보처리기사 실기 기출문제 01회



정보처리기사 실기 시험은 한국산업인력공단에서 문제를 공개하지 않아 문제 복원에 많은 어려움이 있습니다. 다음에 제시된 문제는 시험을 치른 학생들의 기억을 토대로 복원한 것이므로, 일부 내용이나 문제별 배점이 실제 시험과 다를 수 있음을 알립니다.

저작권 안내

이 자료는 시나공 카페 회원을 대상으로 하는 자료로서 개인적인 용도로만 사용할 수 있습니다. 허락 없이 복제하거나 다른 매체에 옮겨 실을 수 없으며, 상업적 용도로 사용할 수 없습니다.

*** 수험자 유의사항 ***

1. 시험 문제지를 받는 즉시 응시하고자 하는 종목의 문제지가 맞는지 확인하여야 합니다.
2. 시험 문제지 총면수·문제번호 순서·인쇄상태 등을 확인하고, 수험번호 및 성명을 답안지에 기재하여야 합니다.
3. 문제 및 답안(지), 채점기준은 일절 공개하지 않으며 자신이 작성한 답안, 문제 내용 등을 수험표 등에 이기 (옮겨 적는 행위) 등은 관련 법 등에 의거 불이익 조치 될 수 있으니 유의하시기 바랍니다.
4. 답안지 작성은 반드시 검정색 필기구만 사용하여야 합니다. (그 외 연필류, 유색 필기구 등을 사용한 답안은 채점하지 않으며 0점 처리됩니다.)
5. 답란(답안 기재란)에는 문제와 관련 없는 불필요한 낙서나 특이한 기록사항 등을 기재하여서는 안되며 부정의 목적으로 특이한 표식을 하였다고 판단될 경우에는 모든 문항이 0점 처리됩니다.
6. 답안을 정정할 때에는 반드시 정정부분을 두 줄(=)로 그어 표시하여야 하며, 두 줄로 긋지 않은 답안은 정정하지 않은 것으로 간주합니다. (수정테이프, 수정액 사용불가)
7. 답안의 한글 또는 영문의 오타자는 오답으로 처리됩니다. 단, 답안에서 영문의 대·소문자 구분, 띄어쓰기는 여부에 관계 없이 채점합니다.
8. 계산 또는 디버깅 등 계산 연습이 필요한 경우는 <문 제> 아래의 연습란을 사용하시기 바라며, 연습란은 채점대상이 아닙니다.
9. 문제에서 요구한 가지 수(항수) 이상을 답란에 표기한 경우에는 답안기재 순으로 요구한 가지 수(항수)만 채점하고 한 항에 여러 가지를 기재하더라도 한 가지로 보며 그 중 정답과 오답이 함께 기재란에 있을 경우 오답으로 처리됩니다.
10. 한 문제에서 소문제로 파생되는 문제나, 가지수를 요구하는 문제는 대부분의 경우 부분채점을 적용합니다. 그러나 소문제로 파생되는 문제 내에서의 부분 배점은 적용하지 않습니다.
11. 답안은 문제의 마지막에 있는 답란에 작성하여야 합니다.
12. 부정 또는 불공정한 방법(시험문제 내용과 관련된 메모지사용 등)으로 시험을 치른 자는 부정행위자로 처리되어 당해 시험을 중지 또는 무효로 하고, 2년간 국가기술자격검정의 응시자격이 정지됩니다.
13. 시험위원이 시험 중 신분확인을 위하여 신분증과 수험표를 요구할 경우 반드시 제시하여야 합니다.
14. 시험 중에는 통신기기 및 전자기기(휴대용 전화기 등)를 지참하거나 사용할 수 없습니다.
15. 국가기술자격 시험문제는 일부 또는 전부가 저작권법상 보호되는 저작물이고, 저작권자는 한국산업인력공단입니다. 문제의 일부 또는 전부를 무단 복제, 배포, 출판, 전자출판 하는 등 저작권을 침해하는 일체의 행위를 금합니다.

※ 수험자 유의사항 미준수로 인한 채점상의 불이익은 수험자 본인에게 전적으로 책임이 있음

문제 1 서브넷 마스크가 255.255.255.0인 네트워크에서 사용 가능한 호스트 ID의 개수는 몇 개인지 쓰시오. (5점)

답 :

문제 2 인덱스 생성 후 운영하는 과정에서 데이터의 Insert, Update, Delete 등의 반복적으로 작업으로 인해 단편화가 발생하는 경우 데이터가 비연속적으로 위치하게 되어 인덱스의 성능이 저하될 수 있다. 이런 경우 다시 순차적인 데이터 저장 형태로 인덱스를 재구성할 필요가 있다. 다음은 idx_T 인덱스의 재구성을 위한 SQL문이다. 괄호에 알맞은 명령어를 쓰시오. (5점)

ALTER INDEX idx_T ()

답 :

문제 3 여러 상태를 갖는 결함을 지속적으로 추적 관리하는 것이 결함 관리의 주요 목적이다. 다음의 결함 관리 과정에서 괄호에 알맞은 단계를 보기에서 찾아 기호로 쓰시오. (5점)

<보기>

㉠ 결함 할당 ㉡ 결함 분석/검토 ㉢ 결함 수정

<결함 관리 프로세스>

결함 등록 → () → () → () → 결함 종료

답 :

문제 4 운영체제에 관한 다음 설명에서 괄호(㉠, ㉡)에 들어갈 알맞은 용어를 쓰시오. (5점)

운영체제는 (㉠) 프로그램과 (㉡) 프로그램으로 나뉜다. (㉠) 프로그램은 컴퓨터 전체의 작동 상태 감시, 작업의 순서 지정, 작업에 사용되는 데이터 관리 등의 역할을 수행하는 것으로 감시 프로그램, 작업 제어 프로그램, 자료 관리 프로그램이 있다. (㉡) 프로그램은 (㉠) 프로그램의 지시를 받아 사용자가 요구한 문제를 해결하기 위한 프로그램으로 언어 번역 프로그램, 서비스 프로그램 등이 있다.

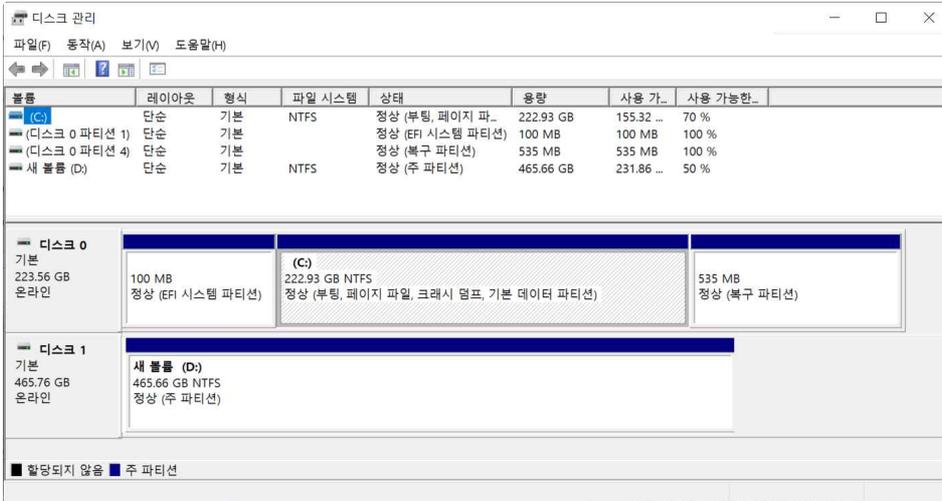
답

- ㉠
- ㉡

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 5 다음은 디스크 관련 작업을 수행하는 디스크 관리 창이다. 다음 그림에 대한 <보기>의 설명 중 옳은 것을 모두 고르시오. (5점)



<보기>

- ㉠ CD-ROM 드라이브가 1개 있다.
- ㉡ 디스크 0은 3개의 파티션으로 구성되어 있다.
- ㉢ D 드라이브의 파일 시스템은 NTFS이다.
- ㉣ 운영체제는 디스크 1에 설치되어 있다.

답 :

문제 6 다음 설명에서 괄호에 들어갈 알맞은 용어를 쓰시오. (5점)

()란 함수적 종속성 등의 종속성 이론을 이용하여 잘못 설계된 관계형 스키마를 더 작은 속성의 세트와 쪼개어 바람직한 스키마로 만들어 가는 과정이다. ()를 거치지 않으면 데이터베이스 내에 데이터들이 불필요하게 중복되어 릴레이션 조작 시 예기치 못한 곤란한 현상이 발생하는데, 이를 이상(Anomaly)이라고 한다. ()를 수행하면 데이터 중복을 배제하여 이상(Anomaly)의 발생 방지 및 자료 저장 공간의 최소화가 가능하다.

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.



문제 7 다음 C 언어로 구현된 프로그램1의 실행 결과가 4일 경우 프로그램2의 실행 결과를 쓰시오. (5점)

프로그램1	프로그램2
<pre>#include <stdio.h> main() { printf("%d", sizeof(12)); return 0; }</pre>	<pre>#include <stdio.h> main() { printf("%d\n", sizeof(12.3)); printf("%d", sizeof(char)); return 0; }</pre>

답 :

문제 8 TCP/IP는 인터넷에 연결된 서로 다른 기종의 컴퓨터끼리 데이터를 주고받을 수 있도록 하는 인터넷 표준 프로토콜이다. TCP/IP 구성 요소 중 문자 형태로 된 도메인 네임을 숫자로 변환해 주는 시스템을 무엇이라고 하는지 영문 약어로 쓰시오. (5점)

답 :

문제 9 릴레이션 구조에 대한 다음 설명에서 괄호(㉠, ㉡)에 들어갈 알맞은 용어를 쓰시오. (5점)

릴레이션은 데이터들을 표(Table)의 형태로 표현한 것으로 구조를 나타내는 릴레이션 (㉠)와 실제 값들인 (㉡)로 구성된다. (㉠)는 릴레이션을 구성하는 속성들의 집합으로, 릴레이션 내포(Intension)라고도 한다. (㉡)는 릴레이션을 구성하는 튜플들의 집합으로 릴레이션의 외포(Extension)라고도 한다.

답

- ㉠
- ㉡

연 습 란

※ 다음 여백은 연습란으로 사용하기 바랍니다.

문제 14 다음은 CPU의 클럭 수를 측정하는 C 프로그램이다. ㉠에 알맞은 헤더파일과 ㉡에 알맞은 함수명을 쓰시오. (5점)

```
#include <stdio.h>
#include "( ㉠ )"
#include "conio.h"

int main(void)
{
    register long i;
    int k;
    clock_t start, finish;
    start = ( ㉡ );
    for (k = 0; k < 100; k++)
        for (i = 0; i < 60000; i++);
    finish = ( ㉡ );
    printf("측정값 = %lf", (float)(finish - start));
}
```

답

- ㉠
- ㉡

문제 15 다음 Java 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (5점)

```
public class Test {
    public static void main(String[] args) {
        int i;
        int num[] = {2, 1, 3, 7, 4, 9};
        int numb[] = new int[10];
        for (i = 0; i < num.length; i++)
            numb[i] = num[i];
        for (i = 0; i < numb.length; i++)
            System.out.printf("%d", numb[i]);
    }
}
```

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 16 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (5점)

```
#include <stdio.h>

int main(void) {
    int i = 200;
    float a = 123.456f;

    i = (int)a;
    printf("%d, %3.2f", i, a);
    return 0;
}
```

답 :

문제 17 다음의 Java 프로그램은 “//오류 발생” 표시 부분과 같이 클래스 외부에서 클래스 변수를 직접 접근하려고 할 때 오류가 발생한다. 이와 같이 클래스 외부에서 클래스 변수 접근을 제어하고자 할 때 코드의 밑줄에 작성해야 하는 알맞은 접근 제어자를 <보기>에서 찾아 쓰시오. (5점)

```
class Data {
    _____ int val = 0;
    void set() { int v = val; }
    int get() { return val; }
}

public class Test {
    public static void main(String[] args) {
        Data d = new Data();
        System.out.printf("%d", d.val); // 오류 발생
    }
}
```

<보기>

private public default protected volatile

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 18 파일이나 폴더를 삭제할 때 [Delete]를 누르면 삭제한 항목이 휴지통에 임시 저장된다. 삭제 시 삭제한 항목이 휴지통에 임시 저장되지 않고 완전히 삭제되도록 하기 위한 다음의 단축키 구성에서 괄호에 들어갈 알맞은 키를 보기에서 찾아 쓰시오. (5점)

() + Delete

<보기>

• Ctrl
• Shift
• Alt
• Windows 키

답 :

문제 19 다음은 <pay> 테이블에 대해 salary가 4000 미만인 직원들의 salary를 4000으로 수정하는 <SQL문>이다. 괄호에 알맞은 명령어를 채워 <SQL문>을 완성하시오. (5점)

<pay> 테이블

id	name	salary	deptid
0001	권을	5000	1
0002	강감찬	3000	3
0003	전봉준	4000	4
0004	이순신	8000	2

<SQL문>

UPDATE pay () salary = 4000 WHERE salary < 4000;

답 :

문제 20 다음 Java로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (5점)

```

public class Test {
    public static void main(String[] args) {
        int num1 = 5, num2 = 7;
        if (++num1 < 5 || ++num2 > 5)
            System.out.println(num1);
        System.out.println(num2);
    }
}
    
```

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.



기출문제 정답

[답안 작성 방법 안내]

‘운영체제(OS; Operation System)’처럼 한글과 영문으로 제시되어 있는 경우 ‘운영체제’, ‘OS’, ‘Operation System’ 중 1가지만 쓰면 됩니다.

[문제 1]

254개

[해설]

255.255.255.0은 C Class의 서브넷 마스크입니다. C Class의 호스트 ID의 개수는 256개이지만 첫 번째 호스트 ID는 네트워크 주소로 사용되고, 마지막 호스트 ID는 브로드캐스트 주소로 사용되므로 실제 사용 가능한 호스트 ID의 개수는 254개입니다.

[문제 2]

REBUILD

[해설]

ALTER INDEX Idx_T	수정할 인덱스의 이름은 'Idx_T'이다.
RUBILD	인덱스를 재구성한다.

[문제 3]

㉠, ㉡, ㉢

[문제 4]

㉡ 제어 ㉠ 처리

[문제 5]

㉠, ㉢

[해설]

㉡ CD-ROM 드라이브는 없고 하드 디스크 드라이브만 2개가 있습니다.

㉢ 디스크 0의 (C:)라고 적힌 파티션의 상태에 '부팅'이 있는 것으로 보아 운영체제는 디스크 0에 설치되어 있음을 알 수 있습니다.

[문제 6]

정규화

[문제 7]

8

1

※ 답안 작성 시 주의 사항

프로그램의 실행 결과는 부분 점수가 없으므로 정확하게 작성해야 합니다. 예를 들어 출력값 사이에 줄바꿈 없이 81 또는 8, 1로 썼을 경우 부분 점수 없이 완전히 틀린 것으로 간주됩니다.

[해설]

• 프로그램1

```
#include <stdio.h>
main() {
  ❶ printf("%d", sizeof(12));
  ❷ return 0;
}
```

❶ 숫자 12는 정수형이므로 정수 자료형인 int의 크기 4를 출력한다.

• sizeof : 자료형의 크기를 바이트(Byte) 단위로 출력

결과 4

❷ 프로그램을 종료한다.

• 프로그램2

```
#include <stdio.h>
main() {
  ❶ printf("%d\n", sizeof(12.3));
  ❷ printf("%d", sizeof(char));
  ❸ return 0;
}
```

❶ 12.3은 실수형이므로 실수 자료형의 크기인 double의 크기 8을 출력한 후 커서를 다음 줄로 처음으로 이동한다.

결과 8

❷ char는 문자 자료형으로 1Byte의 크기를 가지므로 1을 출력한다.

결과 8
1

❸ 프로그램을 종료한다.

[문제 8]

DNS

[문제 9]

㉠ 스키마(Schema) ㉡ 인스턴스(Instance)

[문제 10]

주소 = '서울' and 학과 = '전산'

[해설]

```
SELECT 이름, 학년           '이름'과 '학년' 속성을 표시한다.
FROM 학생                 <학생> 테이블을 검색한다.
WHERE 주소 = '서울' and 학과 = '전산'; '주소'가 "서울"이고 '학과'가 "전산"인 튜플을 대상으로 한다.
```

[문제 11]

관계형 데이터베이스(RDBMS)

[문제 12]

물리 계층(Physical Layer)

[문제 13]

RESTRICT

[문제 14]

㉠ time.h ㉡ clock()

※ 답안 작성 시 주의 사항

C, Java, Python 등의 프로그래밍 언어에서는 대소문자를 구분하기 때문에 대소문자를 구분하여 정확하게 작성해야 합니다. 예를 들어 대문자로 TIME.H나 CLOCK()으로 썼을 경우 부분 점수 없이 완전히 틀린 것으로 간주됩니다.

[해설]

문제의 코드는 ⑤~⑥번 문장을 수행하는 동안 걸린 시간을 CPU의 클럭으로 측정한 후 출력하는 프로그램입니다. CPU의 클럭은 컴퓨터의 환경이나 상태에 따라 달라지므로 결과는 매번 달라질 수 있습니다.

※ 클럭(Clock) : CPU의 주파수가 한 사이클을 도는 데 걸리는 시간으로, 동작 주파수라고도 합니다.

```

#include <stdio.h>           데이터의 입출력에 사용되는 기능들을 제공하는 헤더파일
#include "time.h"           시간 처리에 사용되는 기능들을 제공하는 헤더파일
#include "conio.h"          콘솔 입출력에 사용되는 기능들을 제공하는 헤더파일

int main(void)
{
  ① register long i;
  ② int k;
  ③ clock_t start, finish;
  ④ start = clock();
  ⑤ for (k = 0; k < 100; k++)
  ⑥     for (i = 0; i < 60000; i++);
  ⑦ finish = clock();
  ⑧ printf("측정값 = %lf", (float)(finish - start));
}

```

- ① long형 레지스터 변수 i를 선언한다. 레지스터 변수는 메모리 대신 CPU의 레지스터를 사용하므로 일반 변수보다 속도가 빠르다.
 - register : 레지스터 변수 선언 시 사용하는 예약어
- ② 정수형 변수 k를 선언한다.
- ③ clock_t형 변수 start와 finish를 선언한다.
 - clock_t 자료형 : clock() 함수로 반환되는 값을 저장하기 위한 자료형으로, time.h에 정의되어 있다.
- ④ clock() 함수를 호출한 후 반환되는 값을 start에 저장한다. start에는 프로그램이 실행되었을 때부터 제어가 ④번 문장까지 도달하는 데 걸린 시간이 클럭으로 환산되어 저장된다. (예 : 462.000000)
 - clock() 함수 : 프로그램이 실행되었을 때부터 제어가 함수가 사용된 시점까지 도달하는 데 걸린 시간을 클럭으로 환산하여 반환한다.
- ⑤ 반복 변수 k가 0부터 1씩 증가하면서 100보다 작은 동안 ⑥번을 반복 수행한다.
- ⑥ 반복 변수 i가 0부터 1씩 증가하면서 60000보다 작은 동안 실행 코드 없이 반복 수행한다.
- ⑦ clock() 함수를 호출한 후 반환된 값을 finish에 저장한다. finish에는 프로그램이 실행되었을 때부터 제어가 ⑦번 문장까지 도달하는 데 걸린 시간이 클럭으로 환산되어 저장된다. (예 : 601.000000)
- ⑧ 측정값 = 을 출력하고 finish에서 start를 뺀 값을 실수형으로 변환하여 출력한다. ⑦번 문장까지의 클럭 수에서 ④번 문장까지의 클럭 수를 뺀 값, 즉 ⑤~⑥번 문장의 수행 시간이 클럭으로 환산되어 출력된다. (예 : 139.000000)

[문제 15]

2137490000

[해설]

```

public class Test {
    public static void main(String[] args) {
        ❶ int i;
        ❷ int num[] = {2, 1, 3, 7, 4, 9};
        ❸ int numb[] = new int[10];
        ❹ for (i = 0; i < num.length; i++)
        ❺     numb[i] = num[i];
        ❻ for (i = 0; i < numb.length; i++)
        ❼     System.out.printf("%d", numb[i]);
    }
}

```

- ❶ 정수형 변수 i를 선언한다.
- ❷ 배열을 선언할 때 사용할 개수를 생략하고 초기값을 지정하면, 초기값으로 지정된 값의 수와 같은 크기로 배열이 선언된다.

	num[0]	num[1]	num[2]	num[3]	num[4]	num[5]
배열 num	2	1	3	7	4	9

- ❸ 10개의 요소를 갖는 정수형 배열 numb를 선언한다.

	numb[0]	numb[1]	numb[2]	numb[3]	numb[4]	numb[5]	numb[6]	numb[7]	numb[8]	numb[9]
배열 numb	0	0	0	0	0	0	0	0	0	0

※ Java는 배열 선언 시 초기화를 하지 않아도 자동으로 0으로 초기화됩니다.

- ❹ 반복 변수 i가 0부터 1씩 증가하면서 num 배열의 길이인 6보다 작은 동안 ❺번을 반복 수행한다.
 - **length** : 배열 클래스의 속성으로, 배열 요소의 개수가 저장되어 있다. num 배열은 6개의 요소를 가지므로 num.length는 6을 가지고 있다.
- ❺ num[i]의 값을 numb[i]에 저장한다.

반복문 실행에 따른 변수들의 변화는 다음과 같다.

i	num[i]	numb 배열									
		[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]
0	2	0	0	0	0	0	0	0	0	0	0
1	1	2	0	0	0	0	0	0	0	0	0
2	3	2	1	0	0	0	0	0	0	0	0
3	7	2	1	3	0	0	0	0	0	0	0
4	4	2	1	3	7	0	0	0	0	0	0
5	9	2	1	3	7	4	0	0	0	0	0
6		2	1	3	7	4	9	0	0	0	0

- ❻ 반복 변수 i가 0부터 1씩 증가하면서 numb 배열의 길이인 10보다 작은 동안 ❼번을 반복한다.
 - **numb.length** : numb 배열은 10개의 요소를 가지므로 numb.length는 10을 가지고 있다.
- ❼ numb[i]의 값을 정수형으로 출력한다.

결과 2137490000

[문제 16]

123, 123.46

※ 답안 작성 시 주의 사항

프로그램의 실행 결과는 부분 점수가 없으므로 정확하게 작성해야 합니다. 예를 들어 출력값 사이에 줄을

123

나뉘 123.46 으로 썼을 경우 부분 점수 없이 완전히 틀린 것으로 간주됩니다.

[해설]

```
#include <stdio.h>
int main(void) {
  ❶ int i = 200;
  ❷ float a = 123.456f;
  ❸ i = (int)a;
  ❹ printf("%d, %3.2f", i, a);
  ❺ return 0;
}
```

- ❶ 정수형 변수 i를 선언하고, 200으로 초기화한다. (i = 200)
- ❷ 실수형 변수 a를 선언하고, 123.456으로 초기화한다. (a = 123.456)
- ❸ a의 값을 정수로 변환하여 i에 저장한다. (i = 123)
- ❹ i의 값을 정수로 출력한 후 십표(.)와 공백 한 칸을 출력한다. 이어서 a의 값을 실수로 출력하되 반올림하여 소수점 이하 2자리까지만 출력한다. 전체 3자리를 확보한 후 오른쪽으로 정렬하여 출력해야 하지만 출력할 값인 123.46이 3자리를 넘기 때문에 이를 무시하고 바로 출력된다.
결과 **123, 123.46**
- ❺ 프로그램을 종료한다.

[문제 17]

㉠

[해설]

- 오류가 발생한 코드 `System.out.printf("%d", d.val)`는 Data 클래스 객체 변수 d의 val 변수에 직접 접근하여 값을 출력하는 코드입니다.
- 이 부분에서 오류가 발생했고, 밑줄의 위치를 고려해봤을 때 val 변수로의 접근이 차단되어 오류가 발생했음을 알 수 있습니다.
- 때문에 밑줄에는 클래스 내부에서만 접근할 수 있는 접근 제어자인 `private`가 들어가는 것이 가장 적합합니다.

[문제 18]

Shift

[문제 19]

SET

[해설]

UPDATE pay	<pay> 테이블을 갱신한다.
SET salary = 4000	'salary'의 값을 4000으로 변경한다.
WHERE salary < 4000;	'salary'의 값이 4000 미만인 튜플만을 대상으로 한다.

[문제 20]

6

8

[해설]

```
public class Test {  
    public static void main(String[] args) {  
        ❶ int num1 = 5, num2 = 7;  
        ❷ if (++num1 < 5 || ++num2 > 5)  
            System.out.println(num1);  
        ❸ System.out.println(num2);  
    }  
}
```

- ❶ 정수형 변수 num1과 num2를 선언하고 초기값으로 각각 5와 7을 저장한다.
- ❷ num1의 값이 5보다 작거나 num2의 값이 5보다 크면 ❸번 문장을 수행한다. num1과 num2는 전치증가 연산이므로 1씩 더한 후 조건을 확인한다. num1에 1을 더한 값 6은 5보다 작지 않지만, num2에 1을 더한 값 8은 5보다 크므로 ❸번으로 이동한다.
- ❸ num1의 값 6을 출력한 후 커서를 다음 줄의 처음으로 옮긴다.

결과 6

- ❹ num2의 값을 8을 출력한 후 커서를 다음 줄로 처음으로 옮긴다.

결과 6
8