이 름 :

1. 자바에서는 클래스들을 용도나 목적별로 그룹을 묶어 관리한다. 소스 파일 첫 줄에 기입해야 하는 그룹 생성 구문을 작성하시오.

그룹명 : ncs.test **( )**

1. 자바의 특징을 3개이상 기술하시오.

**①**

**②**

**③**

**④**

1. Java의 Namming Rule에 맞는 것을 모두 고르시오.

Abc, 자바, 1variable, $variable, variable\_, #variable, true, False

# ( )

1. 다음 연산 문장중에서 result의 값을 순서대로 각각 기술하시오.

# ( )

(int var = 10; 인 경우)

**①** result = var++;

**②** result = ++var;

**③** result = var + var/var;

**④** result = var + 1;

1. 다음은 무엇에 대한 설명인가? **( )**

**①** 서로 다른 패키지에 있는 클래스를 참조하기 위함이다.

**②** 소스 파일에 여러 번 선언이 가능하다.

**③** 클래스 선언 이전에 명시한다.

**④** 패키지 선언 다음에 명시한다.

1. 다음 중 형변환을 생략할 수 있는 것을 **2**개 고르시오 **( )**

byte b = 10; float fl = 3; int i = 100; long l = 1000L;

**①** i = (int)b;

**②** fl = (float)b;

**③** fl = (float)3.14;

**④** short s= (short)fl;

1. 자바에서는 현재 작성 중인 클래스와 패키지가 다르면 반드시 포함 선언을 하여야만 다른 클 래스를 사용할 수 있다. 그런데 포함 선언을 하지 않아도 자바 가상머신에 의해 자동 포함 처리되는 패키지명은?

# ( )

1. 현재 작성 중인 클래스는 ncs.smart 패키지에 속해 있다. 현재 클래스에서 app.sample.Book 클래스를 사용하려면 소스 파일에 추가해야 하는 문장은?

# ( )

1. 프로그램을 실행시키는 start up 스위치의 역할을 하는 main() 메소드의 해드(head) 부분을 기술하시오.

# ( )

1. 다음 중 변수 선언시 사용되는 기본(Primitive) 자료형의 할당 크기를 각각 쓰시오. boolean : **( )** byte

int : **( )** byte

float : **( )** byte

short : **( )** byte

char : **( )** byte double : **( )** byte long : **( )** byte

1. 자바에서 클래스명, 패키지명, 메소드명, 변수명 작성시 사용 가능한 기호문자는?

# ( )

1. 다음 연산자 중에 우선 순위가 가장 높은 것은? **( )**

**①** 삼항 연산자

**②** 일반 논리 연산자 &&

**③** 관계 연산자 >=

**④** 쉬프트 연산자 >>>

1. 삼항연산자를 사용하여 문자변수 ch가 가진 값이 대문자인지 물어보는 조건식을 작성하시오. 결과가 참이면 “대문자”, 거짓이면 “대문자 아님” 처리.

String result = **( );**

1. 다음의 문장을 에러 없이 처리되도록 완성하시오. byte b1 = 12;

byte b2 = 15;

**( )** sum = b1 + b2;

1. 다음 실행의 결과값은? ( ) byte b = 125; b += 10;
2. 다음 문장의 실행 결과값은? **( )**

int num = 10;

System.out.println(“result : “ + (12 \* num++));

1. 다음의 문장에 대해 실행 속도를 향상시키기 위한 수정 구문은? int k = 17;

k = k << 3; **( )**

1. 패키지에 대한 설명으로 틀린 것을 고르시오. ( )

① 서로 관련 있는 클래스들끼리 그룹으로 묶어서 효율적으로 관리한다.

② 같은 이름의 클래스라도 다른 패키지에 각각 존재하는 것도 가능 하다.

③ 클래스의 실제 이름은 패키지 명을 포함 하고 있다.

④ 서로 관련 있는 클래스의 소스코드의 묶음 이다.

1. 다음 문장을 import 선언하지 않고 사용할 수 있는 구문으로 변경하시오. import java.util.Scanner; Scanner sc = new Scanner(System.in);

# ( )

1. Sample 클래스의 public static void display(){} 메소드를 다른 클래스에서 사용하기 위한 실행 문장을 기술하시오.

# ( )

1. Java 에서의 배열에 대한 설명으로 틀린 것을 고르시오. ( )

① 다양한 Type 의 여러 Data 를 하나의 이름으로 관리 할 수 있다.

② 많은 양의 데이터를 저장하기 위해서 사용된다.

③ 배열의 사이즈를 선언하지 않으면 사용할 수 없다.

④ 배열은 객체이며 Reference Type 이다.

1. 배열을 선언하거나 초기화 한 것으로 잘못된 것을 2 개 고르시오. **( )**

① int[] arr = {1, 2, 3};

② int[] arr = new int[5];

③ int[] arr = new int[5]{1,2,3,4,5};

④ int arr[5];

1. 다음과 같은 배열이 있을 때 array[3].length 의 값은 얼마인지 쓰시오. ( )

int[][] array = {{ 5, 5, 5, 5, 5},

{ 10, 10, 10},

{ 20, 20, 20, 20},

{ 30, 30}};

1. 배열 array 에 담긴 모든 값을 더하는 프로그램을 완성하시오.

class Test {

public static void main(String[] args){ int[] arr = {10, 20, 30, 40, 50};

int sum = 0;

for(int i=0;i<arr.length;i++) {

}

System.out.println("sum="+sum);

}

}

1. 다음 프로그램의 실행 결과를 쓰시오. ( )

public class MultiArrayTest1 {

public static void main(String[] args){ int[] no={100,97,89}; int[][]count={{3,4,6},{7,4,5}};

System.out.println(no.length); System.out.println(count.length); System.out.println(count[1].length); System.out.println(no[2]+count[1][2]);

}

}