[2. 변수와 타입]

1. 다음 표에 맞는 데이터 타입(int, boolean, char 등..)을 채워넣으시오

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1byte | 2byte | 4byte | 8byte |
| 논리형 | boolean | X | X | X |
| 문자형 | X | string | X | X |
| 정수형 | byte | short char | int | long |
| 실수형 | X | X | float | double |

1. 타입변환시 허용범위가 큰 순에서 작은순으로 나열하시오.

double –float - long – int – short,char - byte

1. 다음의 출력결과를 예상 후 코드를 실행하시오.

System.out.println(“1” + “2”) → ( “12”)

System.out.println(‘A' + 'B') → ( “131”)

System.out.println('1' + 2) → ( ”52”)

System.out.println('1' + '2') → ( “99’”)

System.out.println('J' + “ava”) → ( “Java”)

1. 다음 자동 타입 변환에서 컴파일 에러가 발생하는 것을 선택하세요. -> 3

|  |
| --- |
| byte byteValue = 10;  char charValue = ‘A’; |

➊ int intValue = byteValue;

➋ int intValue = charValue;

➌ short shortValue = charValue;

➍ double doubleValue = byteValue;

1. 다음 강제 타입 변환에서 컴파일 에러가 발생하는 것을 선택하세요. ->

|  |
| --- |
| int intValue = 10;  char charValue = 'A';  double doubleValue = 5.7;  String strValue = "A"; |

➊ double var = (double) intValue;

➋ byte var = (byte) intValue;

➌ int var = (int) doubleValue;

➍ char var = (char) strValue;

1. 연산식의 타입 변환 중에서 컴파일 에러가 발생하는 것을 선택하세요.

|  |
| --- |
| byte byteValue = 10;  float floatValue = 2.5F;  double doubleValue = 2.5 |

➊ byte result = byteValue + byteValue;

➋ int result = 5 + byteValue;

➌ float result = 5 + floatValue;

➍ double result = 5 + doubleValue;

1. 문자열을 기본 타입으로 변환하는 코드로 틀린 것을 고르세요. -> 2

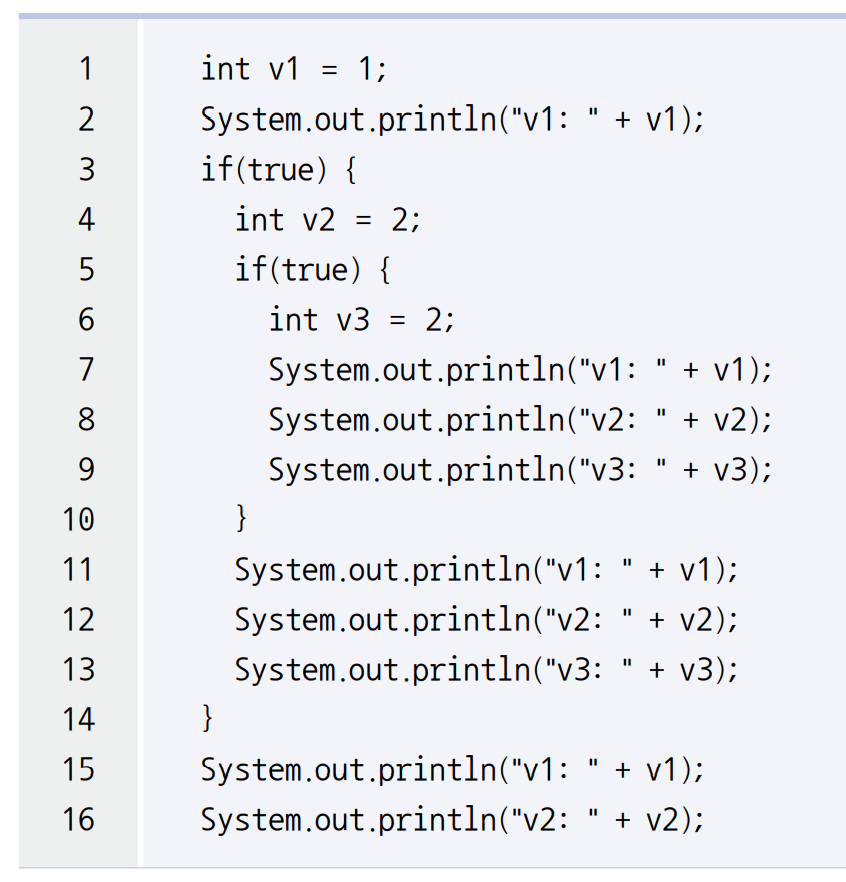
|  |
| --- |
| String str = “5”; |

➊ byte var1 = Byte.parseByte(str);

➋ int var2 = Int.parseInt(str);

➌ float var3 = Float.parseFloat(str);

➍ double var4 = Double.parseDouble(str)

1. 다음 코드에서 컴파일 에러가 나는 라인을 모두 적어보세요. 

16번, 13번

1. 다음의 문장에서 리터럴, 변수, 키워드를 구분하여 적으시오.

int a = 20;

double num = 1.2;

char ch = ‘a’;

-리터럴: 20, 1.2 ‘a’

-변수: a num ch

-키워드: int double char

1. 다음 중 변수를 잘못 초기화 한 것은 -> 3

➊ byte b = 256;

➋ char c = '';

➌ char answer = 'no';

➍ double d = 1.4e3f;

1. 다음 중 형변환을 생략할 수 있는 것은? -> 2

|  |
| --- |
| byte b = 10;  char ch = 'A';  int i = 100;  long l = 1000L; |

➊ b = (byte)i;

➋ ch = (char)b;

➌ short s = (short)ch;

➍ i = (int)ch;