

2019 JAVA 중간 고사

학번: _____

이름: _____

1. 자바 언어가 가진 특징 중 "플랫폼 독립성"이란 무엇인지 간단히 설명하고, 자바는 이를 어떤 방식으로 실현했는지 간단히 설명하라.

플랫폼 독립성:

자바의 실현방식:

2. 자바 응용프로그램의 실행을 위한 main 함수가 정적으로 정의되어 있는 이유는 무엇인가?

3. 자바의 자료형은 크게 기초형과 참조형으로 나뉜다. 둘의 차이에 대해 간략히 설명하라.

4. 다음 C의 main 함수와 같은 역할을 하는 java 코드를 작성하라. (단 파일명은 Test.java이다)

```
//-- test.c 파일 --  
  
#include "stdio.h"  
  
void main()  
{  
    int a;  
    scanf("%d", &a);  
}
```

```
// -- Test.java 파일 --
```

5. 다음은 JAVA 코드의 일부분이다. 틀린 부분을 모두 찾아 오른쪽 칸에 고쳐 쓰라.

```
float a = 10.0;  
int b = a / 3;  
byte c = b;  
boolean d = (c>3)?1:0;
```

6. 자바에서 소멸자(destructor)는 명확히 존재하진 않으며 (있을지라도) 사용이 권장되지 않는 이유는 무엇인가?

--

7. 다음과 같은 비정방형 2차원 int 배열 arr을 생성하는 코드를 작성하라. (단, class선언 등을 포함한 코드 전체가 아닌, 배열 생성 부분만 제시할 것)

<table border="1"><tr><td>arr[0][0]</td></tr><tr><td>arr[1][0]</td><td>arr[1][1]</td></tr><tr><td>arr[2][0]</td><td>arr[2][1]</td><td>arr[2][2]</td></tr><tr><td>arr[3][0]</td><td>arr[3][1]</td><td>arr[3][2]</td><td>arr[3][3]</td></tr></table>	arr[0][0]	arr[1][0]	arr[1][1]	arr[2][0]	arr[2][1]	arr[2][2]	arr[3][0]	arr[3][1]	arr[3][2]	arr[3][3]	code:
arr[0][0]											
arr[1][0]	arr[1][1]										
arr[2][0]	arr[2][1]	arr[2][2]									
arr[3][0]	arr[3][1]	arr[3][2]	arr[3][3]								

8. 다음은 입력 받은 배열의 수를 모두 더한 값을 계산하여 반환하는 함수이다. 함수의 중간 빈 부분을 완성하라.

<pre>int sum(int [] arr) { int out = 0; return out; }</pre>
--

9. 다음은 주어진 3개의 값 a,b,c 중 가장 큰 값을 계산하는 함수이다. 다음 빈 칸()에 "3항연산자"를 이용하여 한 문장으로 이를 완성하라.

<pre>int getMax(int a, int b, int c) { return _____; }</pre>
--

10. 객체지향언어의 특성(설계개념) 3가지의 이름을 쓰라.

1	2	3
---	---	---

11. 다음 코드의 실행 출력 결과를 예상하여 보라

<pre>String a = "Sejong"; String b = a; a += "University"; System.out.println(b);</pre>	출력결과:
---	-------

12. 다음 코드의 잘 못된 부분을 찾아 이유를 설명하라.

<pre>class Count{ static int sum = 0; int num = 1; static void setSum(){ sum += num; } }</pre>	잘못된 부분과 이유:
--	-------------

13. 다음 주어진 Class A와 B가 있을 때 "B b = new B(11)"을 실행했을 때의 출력값은 무엇인가?

<pre>class A{ public A() { System.out.println("A"); } public A(int x) { System.out.println("A:"+x); } } class B extends A { public B() { super(100); } public B(int x) { System.out.println("B:"+x); } }</pre>	출력값:
--	------

14. 주어진 인터페이스 Edible를 구현하여 Apple class를 만들려고 한다. 빈 칸()을 채워 최소한의 코드를 완성하여 보라.

<pre>interface Edible { void eat(); }</pre>	<pre>class Apple { { System.out.println("Eat Apple!"); } }</pre>
---	--

15. 다음 빈칸에 들어간 내용을 채워서 Child 클래스의 print() 호출 시 Parent 클래스의 print() 역시 호출 될 수 있도록 빈 칸()을 채워 변경해보라.

<pre>class Parent{ void print(){ System.out.println("parent") } } class Child extends Parent{ void print() { System.out.println("child"); } }</pre>

- 수고하셨습니다. -