

기술인증테스트

Java개발-A

[2017년 @차]

사번 : _____

성명 : _____

1. 자사의 표준 프레임워크는 Spring 기반의 DevOnFrame 이다. 프로젝트에서 DevOnFrame 을 적용할 때의 특징이 아닌 것을 고르시오.

- ① 프레임워크에서 제공하는 기능들은 시스템 개발(SI)에 집중되어 있어, 유지보수(SM)에서는 효과를 얻기 어렵다.
- ② 프레임워크를 사용하면 개발 초기에는 Learning Curve 로 인해 생산성이 저하될 수도 있다.
- ③ 프레임워크를 적용하면 웹 취약점(OWASP 등)을 다수 해결할 수 있다.
- ④ 프레임워크를 적용하면 업무개발자는 비즈니스 로직 개발에 집중 할 수 있다.

(정답) 1

(해설) 프레임워크를 사용하면 유지보수가 용이하다.

2. 아래 프로그램의 실행 결과를 작성하시오.

```
public class NotebookTest {

    public static void main( String[] args ) {

        int variable = 10;
        NotebookTest nt = new NotebookTest();
        Notebook notebook = new Notebook( "Notebook", 150 );

        nt.operator( notebook, variable );

        System.out.println( notebook + "," + variable );
    }

    public void operator( Notebook notebook, int variable ) {
        notebook.price = 100;
        variable = 20;
    }
}

class Notebook {

    String notebookModel;
    int price;
```

```

public Notebook( String notebookModel, int price ) {
    this.notebookModel = notebookModel;
    this.price = price;
}

public String toString() {
    return notebookModel + price;
}
}

```

(정답) Notebook100,10

(해설) 객체를 파라미터로 넘길 경우 객체의 멤버변수에 대한 접근이 가능하다.

3. 아래 프로그램에 대한 설명으로 옳은 것을 고르시오.

```

public class ExceptionTest {

    public static void main( String[] args ) {

        try {
            throw new NoDataBizException();
        } catch ( BizException be ) {    // (A)
            System.out.println( "BizException caught" );
        } catch ( Exception e ) {        // (B)
            System.out.println( "Exception caught" );
        }
    }
}

```

class BizException extends Exception { }

class NoDataBizException extends BizException { }

① Exception caught 가 출력된다.

- ② BizException caught 가 출력된다.
- ③ (A) 라인에서 컴파일 에러가 발생한다.
- ④ (B) 라인에서 컴파일 에러가 발생한다.

(정답) 2

(해설) Exception 의 하위에서부터 상위순으로 처리한다.

4. 다음 중 추상 클래스에 대한 설명으로 틀린 것을 고르시오.

- ① 생성자, 멤버 변수, 추상 메서드를 가질 수 있다.
- ② class 선언부에 abstract 키워드를 붙인다.
- ③ 추상 메서드 없이 추상 클래스가 될 수 있다.
- ④ 일반 클래스가 추상 메서드를 가질 수 있다.

(정답) 4

(해설) 일반 클래스는 추상 메서드를 가질 수 없다.

5. 다음 코드의 실행 결과를 기술하시오.

```
public class Test {  
    public static void main(String[] args) {  
        int [] x = {1,2,3};  
        int y [] = {4,5,6};  
        new Test().method(x,y);  
    }  
    void method(int[]... b) {  
        for(int[] a : b) {  
            System.out.print(a[0]);  
        }  
    }  
}
```

(정답) 14

(해설) 가변인자를 사용하는 메서드의 호출에서 b 변수는 이차원 정수 배열(x,y)을 인자로 받게 된다.

따라서, method(int[] ... b)의 향상된 for 문은 정수 배열 x, y 실행한다.

x[0]=1, y[0]=4 이므로 출력은 14 가 된다.

6. 아래 코드를 실행하였더니 다음 StackTrace 를 남기는 NullPointerException 이 발생하였다. 아래 코드의 20 번째 줄에서 Exception 이 발생한 원인에 대한 설명으로 옳은 것을 고르시오.

----- StackTrace -----

Exception in thread "main" java.lang.NullPointerException

at portal.Test1.test(Test1.java:20)

at portal.Test1.main(Test1.java:13)

```

16:    private static void test( Item item ) {
17:
18:        Manager manager = item.getManager();
19:
20:        List<Menu> children = getMenuItems( item.getManager().getId(), item.getName() );
21:
22:        for ( Menu child : children ) {
23:            deleteMenuItem( child );
24:        }
25:    }
26:
27: private static List<Menu> getMenuItems( int managerId, String name ) {
28:    ...
29: }
```

- ① item 이 null 을 반환하기 때문이다.
- ② item.getManager() 이 null 을 반환하기 때문이다.
- ③ item.getManager().getId() 가 null 을 반환하기 때문이다.
- ④ item.getName() 이 null 을 반환하기 때문이다.

(정답) 2

7. 아래 제시된 프로그램은 18 line 에서 컴파일 오류가 발생한다.

컴파일 오류가 발생하지 않도록 소스코드를 변경 시 적절한 것을 고르시오.

```
1: public class JavaTest {
2:     public static void main(String[] args) {
3:         Child c = new Child(1, 2);
4:         c.print();
5:     }
6: }
7: class Parent {
8:     int a;
9:     Parent(int a) {
10:         this.a = a;
11:     }
12:     void print() {
13:         System.out.println("this.a=" + this.a);
14:     }
15: }
16: class Child extends Parent {
17:     int b;
18:     Child(int a, int b) {      <-- 문항을 CBT 로 만들 때에 해당 라인에 밑줄넣어 주세요.
19:         this.a = a;
20:         this.b = b;
21:     }
22:     void print() {
23:         super.print();
24:         System.out.println("this.b=" + this.b);
25:     }
26: }
```

1) 19 line 을 this(a); 로 변경한다.

2) 19 line 을 super(a); 로 변경한다.

3) 19 line 을 `super.a = a;` 로 변경한다.

4) Child class 에 디폴트 생성자를 추가한다.

(정답) 2

(해설) 컴파일러가 자식클래스 생성자의 첫줄에 `super()`를 추가하지만

부모클래스에 기본생성자 `Parent()`가 존재하지 않기 때문에 컴파일 오류가 발생한다.

수고하셨습니다.



Copyright © 2017 by LG CNS All rights reserved.