

기술인증테스트

Java개발-A

[2017년 #차]

사번 : _____

성명 : _____

1. 자사의 표준 프레임워크는 Spring 기반의 DevOnFrame 이다. 프로젝트에서 프레임워크의 적용 전략으로 맞지 않은 것을 고르시오.

- ① 개발자들의 Learning Curve 를 고려하여 초기 개발 계획을 수립해야 한다.
- ② 프로젝트에서는 프레임워크에서 제공하는 기능 중 필요한 영역을 선별하여 적용해야 한다.
- ③ 프로젝트에서는 필요 시, Customizing 을 통해 최적화하여 사용할 수 있다.
- ④ 프레임워크의 개발 가이드(WIKI 등)를 개발 표준으로 그대로 사용할 수 있다.

(정답) 4

(해설) 프레임워크의 개발가이드는 사용법 위주이며, 프로젝트 표준에 맞게 개발가이드를 개발하여야 한다.

2. 아래 프로그램의 실행 결과를 적으시오.

(※ 대소문자와 White Space 를 고려하여 정확하게 작성하여야 한다.)

```
public class StringCompare {  
  
    public static void main(String[] args) {  
  
        String str1 = new String("java");  
        String str2 = new String("java");  
        StringBuffer sb = new StringBuffer("java");  
  
        if (str1 == str2) {  
            System.out.print("A");  
        }  
  
        if (str1.equals(str2)) {  
            System.out.print("B");  
        }  
  
        if (str1.equals(sb)) {  
            System.out.print("C");  
        }  
    }  
}
```

(정답) B

(해설) String 의 값 비교는 equals 를 이용해야 한다.

3. 다음 프로그램의 실행 결과로 적절한 것을 고르시오.

```
public class ExceptionTest {  
  
    public static void main( String[] args ) {  
  
        int total = -1;  
  
        try {  
            total = sum();  
            System.out.println( "total = " + total );  
        } catch ( RuntimeException e ) {  
            System.out.println( "exception at main()" );  
        }  
    }  
  
    private static int sum() {  
  
        int sum = 0;  
        int array[] = { 10, 20, 40 };  
  
        try {  
            for ( int i = 0; i <= array.length; i++ ) {  
                sum += array[i];  
            }  
        } catch ( ArrayIndexOutOfBoundsException e ) {  
            System.out.println( "exception at sum()" );  
            throw new RuntimeException();  
        } finally {  
            System.out.println( "return from sum()" );  
            return sum;  
        }  
    }  
}
```

```

    }
}

```

①

```

exception at sum()
exception at main()

```

②

```

return from sum()
total = 70

```

③

```

exception at sum()
return from sum()
total = 70

```

④

```

exception at sum()
return from sum()
exception at main()

```

(정답) ③

(해설) try 절에서 return 하더라도 finally 는 수행된다.

4. 회원정보를 변경하는 메소드를 포함한 프로그램을 아래와 같이 작성하였다.
main() 메소드의 수행 결과를 작성하시오.

```

public class MemberAgeChange {
    public static void main(String[] args) {
        MemberAgeChange mac = new MemberAgeChange();

        int age = 30;
        Member member = new Member("JAMES", 30);
        mac.change(member, age);

        System.out.print(member + "," + age);
    }
}

```

```

    public void change(Member member, int age) {
        member.age = 50;
        age = 40;
    }
}

```

```

class Member {
    String name;
    int age;

    public Member(String name, int age) {
        this.name = name;
        this.age = age;
    }

    @Override
    public String toString() {
        return name + age;
    }
}

```

(정답) JAMES50,30

(해설) product 는 reference 로, variable 은 value 로 전달됨

5. 프로그램 실행 중 에러가 발생하여 디버깅을 수행하는 중이다. 서버의 로그 파일에서 아래와 같은 StackTrace 가 발견되었다. 이 StackTrace 를 통해 파악할 수 있는 사항으로 옳지 않은 것을 고르시오.

```

java.io.IOException: Unexpected exeption on report manager
    at com.sales.process.ReportService.retrieveReportMaster(ReportService.java:229)
    at com.sales.process.ReportController.retrieveOrderSummaryReport(ReportController.java:214)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:483)
    at org.springframework.web.servlet.DispatcherServlet.doDispatch(DispatcherServlet.java:959)
    at org.springframework.web.servlet.DispatcherServlet.doService(DispatcherServlet.java:893)

```

```

at org.springframework.web.servlet.FrameworkServlet.doGet(FrameworkServlet.java:860)
at org.springframework.web.servlet.FrameworkServlet.service(FrameworkServlet.java:845)
at javax.servlet.http.HttpServlet.service(HttpServlet.java:731)
at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:303)
at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:208)
at org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:52)
at org.springframework.web.filter.CharacterEncodingFilter.doFilterInternal(CharacterEncodingFilter.java:121)
at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:208)
at org.springframework.web.filter.DelegatingFilterProxy.doFilter(DelegatingFilterProxy.java:262)
at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:208)
at org.apache.catalina.valves.AccessLogValve.invoke(AccessLogValve.java:956)
at org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:423)
at org.apache.coyote.http11.AbstractHttp11Processor.process(AbstractHttp11Processor.java:1079)
at org.apache.tomcat.util.net.JIoEndpoint$SocketProcessor.run(JIoEndpoint.java:318)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
at org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)
at java.lang.Thread.run(Thread.java:745)
Caused by: java.io.FileNotFoundException: Unexist File OrderReportBuilder.rep
at com.sales.process.ReportService$OrderReportManager.workWithFile(ReportService.java:112)
at com.sales.process.ReportService.retrieveReportMaster(ReportService.java:203)
... 63 more

```

- ① 위 프로그램은 apache tomcat 이 생성한 thread pool 에서 thread 를 할당받아 수행되었다.
- ② IOException 이 발생한 원인은 ReportService 의 112 라인에서 발생한 FileNotFoundException 이다.
- ③ ReportService 는 적어도 하나의 try~catch 문을 가지고 있다.
- ④ OrderReportManager 는 ReportService 를 상속받은 클래스이다.
- ⑤ ReportController 의 retrieveOrderSummaryReport() 메소드 214 라인에서 ReportService 의 retrieveReportMaster() 메소드를 호출하였다.

(정답) ④

(해설) ④ OrderReportManager 는 ReportService 의 inner class 이다.

6. XML 문서를 해독하고 유효한 데이터를 뽑아오기 위해 DOM 또는 SAX 파서를 사용한다.

특히, XML 문서의 일부분이 필요하거나 빠른 속도로 XML 문서를 파싱하고자 할 때는 SAX 파서를 주로 사용한다.

아래 프로그램은 SAX 파서를 사용한 예이다. main 메소드의 실행 결과 중 올바른 것을 고르시오.

```

public class XmlParser {
    public static void main(String[] args) throws Exception {
        SAXParserFactory factory = SAXParserFactory.newInstance();
        SAXParser parser = factory.newSAXParser();
        parser.parse(new File("list.xml"), new ListHandler());
    }
}

class ListHandler extends DefaultHandler {

    @Override
    public void startElement(String uri, String localName, String qName,
        Attributes attributes) throws SAXException {
        System.out.println("STA " + qName);
    }

    @Override
    public void endElement(String uri, String localName, String qName)
        throws SAXException {
        System.out.println("END " + qName);
    }
}

```

파싱에 사용한 list.xml 파일의 내용은 아래와 같다.

```

<?xml version="1.0"?>
<list>
    <department id="1">
        <name>CTO</name>
        <number>80913</number>
    </department>
</list>

```

①

STA list

END list

STA department

END department

STA name

END name

STA number

END number

②

STA list

STA department

STA name

END name

STA number

END number

END department

END list

③

STA /list

STA /list/department

STA /list/department/name

END /list/department/name

STA /list/department/number

END /list/department/number

END /list/department

END /list

④

STA /list

END /list

STA /list/department

END /list/department

STA /list/department/name

END /list/department/name

STA /list/department/number

END /list/department/number

(정답) ②

(해설) SAX 파서는 xml 태그를 순서대로 읽어들인다.

7. 아래 프로그램의 실행 결과를 적으시오.


```
class Product {
    public int productNumber = 9;
    void showProductType() {
        System.out.print( "P" );
    }
}

class Radio extends Product {
    public int productNumber = 5;
    void showProductType() {
        System.out.print( "R" );
    }
}

class SmartRadio extends Radio {
    public int productNumber = 3;
    void showProductType() {
        System.out.print( "S" );
    }
}

public class Exercise {

    public static void main( String[] args ) {

        SmartRadio smartRadio = new SmartRadio();

        Radio radio = smartRadio;
        Product product = radio;

        product.showProductType();

        System.out.print( radio.productNumber );
        System.out.print( smartRadio.productNumber );
    }
}
```

(정답) S53

(해설) 멤버 변수는 오버라이딩 되지 않는다.

수고하셨습니다.



Copyright © 2017 by LG CNS All rights reserved.