

JAVA 프로그래밍 – 학기말고사 1/2

학년: 학번: 이름: .

① ①②③④⑤⑥ ①②③④⑤⑥⑦⑧⑨⑩⑪⑫⑬⑮

참고 :

1. 개념 작성 문제의 경우 큰 흐름상 이상이 없으면 정답, 단 굵은 글씨체의 키워드 혹은 유사 개념이 포함하여야 함
2. 소스를 작성하는 문제의 경우 참고의 정의가 있는 경우 정확히 기입하여야 정답, 그렇지 않는 경우 개념만 맞으면 정답
3. 소스를 작성하는 문제의 경우 공개된 정답에 의하여서만 채점, 본인 코드가 정확하다고 생각되면 스스로 컴파일하여 동작여부를 증명하여 점수를 획득 할 것

1. 프로세스(Process)와 스레드(Thread)에 대하여 간략히 설명하시오. (10점)

프로세스(Process)

실행 중인 **프로그램 단위**. 컴퓨터에서 프로그램을 실행하면 적어도 프로세스를 한 개 이상 동작하게 됨

스레드(Thread)

프로세스에 있는 리소스(resource)를 공유하는 프로그램의 **실행 단위**
프로세스에서 독자적인 명령어 제어권을 가짐
단일 프로세스에서 복수개의 스레드가 존재할 수 있음

2. 스레드(Thread)를 사용하는 프로그램에서 발생 할 수 있는 문제점(2가지 이상)과 해결 방법(문제점에 대한 해결, 2가지 이상)에 대하여 간략히 설명하시오. (10점)

1. 스레드 접근변수의 라이프 사이클이 끝나는 문제

스레드의 인스턴스를 생성하는 변수를 로컬변수로 선언했을 때
선언된 함수를 종료하면 선언된 변수 또한 메모리에서 사라짐
스레드는 메모리에 존재하나 이 스레드를 다시 제어할 수 없음

해결책 : 스레드를 생성하는 변수를 필드변수로 선언

스레드의 우선권 문제(경쟁상태)

우선권이 낮은 스레드는 작업을 종료하지 못하기도 함

공용 리소스의 접근 문제(교착상태)

여러 스레드가 동시에 접근 시 교착 상태에 빠지게 될 수가 있음

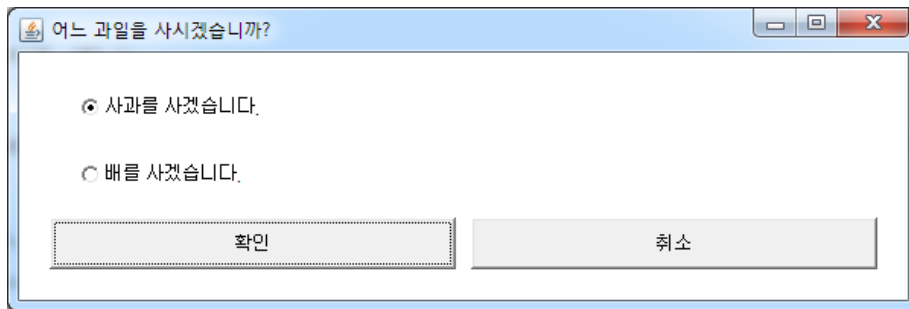
문제를 효과적으로 해결하는 방법

동기화 기술을 사용

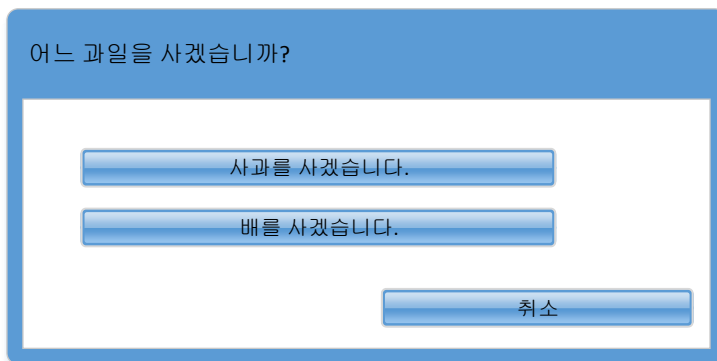
교착 상태를 해결하는 기술

자원에 상호배타적인 권리를 제공해서 교착 상태를 해결
한 스레드에 자원이 있으면 다른 스레드는 그 스레드가 자원을 내놓을 때까지
대기하도록 만듦

3. 아래의UI를 마우스 클릭횟수 등에 대하여 조금 더 효율적으로 개선할 수 있는 방안을 간략한
그림으로 제시하고 동작 과정을 설명하시오. (10점)



정답 :



원래의 UI는 사과를 선택하건 배를 선택하건 라디오 버튼 한번과 확인 버튼을 한번 누르는 과정
이 필요하고, 취소는 한번의 클릭으로 가능함. 전체 경우의 수 5번 클릭
개선된 UI는 사과를 선택하건 배를 선택하건 취소를 하건 모두 한번의 클릭으로 선택 가능함. 전
체 경우의 수 3번 클릭

제시된 정답 이외에 개선 효과가 전체 경우의 수가 마우스 클릭 한번이라도 줄일 수 있으면 정답.
이외의 모든 고민 흔적은 5점.

4. 상기 [문제 3]의 그림으로 제시된 UI 를 생성할 수 있는 JAVA 코드를 기술하시오. (단, 윈도우의
크기는 가로 600, 세로 300으로 설정) (10점)

```
public class GuiAwtTest extends Frame {  
    private Checkbox check1;  
    private Checkbox check2;  
    private CheckboxGroup checkGrp;  
    private Button btn1;  
    private Button btn2;  
  
    public void run() {
```

[문제 3]의 그림과 같이 GUI를 구성하기 위한 코드 작성

```
}  
  
public static void main(String[] args) {  
    // TODO Auto-generated method stub  
    GuiAwtTest gui = new GuiAwtTest();  
    gui.run();  
}  
  
}
```

정답

```
package FinalExam;  
import java.awt.BorderLayout;  
import java.awt.Button;  
import java.awt.Checkbox;  
import java.awt.CheckboxGroup;  
import java.awt.Frame;  
import java.awt.GridLayout;  
import java.awt.Panel;  
  
public class GuiAwtTest extends Frame {  
    private Checkbox[] check;  
    private CheckboxGroup checkGrp;  
    private Button[] btn;  
  
    public void run() {  
        setTitle("어느 과일을 사시겠습니까?");  
        setSize(600, 200);  
  
        // checkbox 생성 및 그룹화  
        check = new Checkbox[2];  
        checkGrp = new CheckboxGroup();  
        check[0] = new Checkbox("사과를 사겠습니다.", checkGrp, true);  
        check[1] = new Checkbox("배를 사겠습니다.", checkGrp, false);  
  
        // button 생성  
        btn = new Button[2];  
        btn[0] = new Button("확인");  
        btn[1] = new Button("취소");  
  
        Panel c1Panel = new Panel( new BorderLayout(10, 10) );  
        c1Panel.add("West", new Panel()); // 왼쪽 가장자리 남기기 위해 추가  
        c1Panel.add("Center", check[0]); // 사과 checkbox 추가
```

```

        Panel c2Panel = new Panel( new BorderLayout(10, 10) );
        c2Panel.add("West", new Panel());           // 왼쪽 가장자리 남기기 위해 추가
        c2Panel.add("Center", check[1]); // 배 checkbox 추가

        Panel cPanel = new Panel( new GridLayout(3, 2, 10, 10) );
        cPanel.add(c1Panel);
        cPanel.add(new Panel());           // 빈공간 유지 위해 추가
        cPanel.add(c2Panel);
        cPanel.add(new Panel());           // 빈공간 유지 위해 추가
        cPanel.add(btn[0]);
        cPanel.add(btn[1]);

        setLayout(new BorderLayout(10, 10));
        add("North", new Panel());         // 위쪽 가장자리 남기기 위해 추가
        add("West", new Panel());          // 왼쪽 가장자리 남기기 위해 추가
        add("East", new Panel());          // 오른쪽 가장자리 남기기 위해 추가
        add("South", new Panel());         // 아래쪽 가장자리 남기기 위해 추가
        add("Center", cPanel);             // 실제 내용 추가

        setVisible(true);

        // 타이틀바의 종료버튼 눌렀을 때 종료 하는 기능 추가
        addWindowListener(new java.awt.event.WindowAdapter() {
            public void windowClosing(java.awt.event.WindowEvent e) {
                dispose();
            }
        });
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        GuiAwtTest gui = new GuiAwtTest();
        gui.run();
    }
}

```

※아래 소스코드를 참고하여 물음에 답하십시오. (5번~7번 문제)

```

package java_book;

①_____//라이브러리 추가
②_____//라이브러리 추가

public class socket_client{
    public static final int BUFFER_SIZE = 1024;

    public static void main(String[] args) {

```

```

String strData =
    "③ _____"
    + "Host: JavaPrograming\n"
    + "User-Agent: Mozilla/4.0 (compatible: MSIE 7.0; Windows NT 5.1)\n"
    + "\n";
byte[] buffer = new byte[BUFFER_SIZE];
④ _____; //Host의 주소 선언 및 정의
⑤ _____; //Host의 접속 포트 선언 및 정의
Socket soHttp = null;

try{
    ⑥ _____; //Socket의 인스턴트 생성
    ⑦ _____; //OutputStream 획득
    ⑧ _____; //DataOutputStream의 인스턴트 생성
    ⑨ _____; //HTTP를 통해 Host로 데이터 전송

    InputStream is = soHttp.getInputStream();
    DataInputStream dis = new DataInputStream(is);
    int nOff = 0;
    int nRead = 0;

    do{
        nRead = dis.read(buffer, nOff, BUFFER_SIZE);

        String strRead = new String(buffer);
        System.out.print( strRead );
    }while(nRead == BUFFER_SIZE);

    ⑩ _____; //Socket 닫기

}catch(Exception e)
{
    e.printStackTrace();
}
}

```

① ② ③ ④ ⑤ ⑥ ⑦ ⑧ ⑨ ④ ⑩ ⑪ ⑫ ⑬ ⑮

5. ①,②에 알맞은 라이브러리를 추가하시오.(10점)

정답:

```

import java.io.*;
import java.net.*;

```

Char 단위로 틀린 점이 발견 되면 -1점 감점

6. ③에 알맞은 아래의 URL의 그림파일을 socket 통신을 이용하여 획득하기 위한 GET방식의 HTTP 요청 문자열을 아래 표를 참고하여 완성하시오. (10점)

그림 파일 url : http://klas.khu.ac.kr/webdata/ko/images/main/img_main_roll_01.jpg

참고 표)

url	http://klas.khu.ac.kr/
-----	---

ip	163.180.96.213
http port	80
http request	"GET / HTTP/1.1\n Host: JavaPrograming\n User-Agent: Mozilla/4.0 (compatible: MSIE 7.0; Windows NT 5.1)\n \n"

정답 :

GET /webdata/ko/images/main/img_main_roll_01.jpg HTTP/1.1\n

Char 단위로 틀린 점이 발견 되면 -1점 감점

7. 소켓 통신을 이용하여 데이터를 획득하는 과정에 대한 ④~⑩의 알맞은 코드를 완성하시오.(10점)

아무런 코드가 없으면 0점

타입, 함수 및 파라미터 등의 잘못된 사용이 발견 되면 -2점

정답 :

```
package java_book;

//라이브러리 추가
import java.io.*;
import java.net.*;

public class socket_client{
    public static final int BUFFER_SIZE = 1024;

    public static void main(String[] args) {

        String strServer = "163.180.96.213";//Or "klas.khu.ac.kr";//Host 선언 및 정의
        int nPort = 80;//Host의 접속 포트 선언 및 정의
        Socket soHttp = null;
        String strData =
            "GET /webdata/ko/images/main/img_main_roll_01.jpg HTTP/1.1\n"
            + "Host: JavaPrograming\n"
            + "User-Agent: Mozilla/4.0 (compatible: MSIE 7.0; Windows NT 5.1)\n"
            + "\n";
        byte[] buffer = new byte[BUFFER_SIZE];

        try{

            soHttp = new Socket( strServer, nPort );//Socket의 인스턴트 생성
            OutputStream os = soHttp.getOutputStream();//OutputStream 획득
            DataOutputStream dos = new DataOutputStream(os);//DataOutputStream의
            인스턴트 생성

            dos.writeBytes(strData);//HTTP를 통해 Host로 데이터 전송

            InputStream is = soHttp.getInputStream();
            DataInputStream dis = new DataInputStream(is);
```

```

        int nOff = 0;
        int nRead = 0;

        do{
            nRead = dis.read(buffer, nOff, BUFFER_SIZE);

            String strRead = new String(buffer);
            System.out.print( strRead );
        }while(nRead == BUFFER_SIZE);

        soHttp.close();//Socket 닫기

    }catch(Exception e)
    {
        e.printStackTrace();
    }
}

```

※아래 소스코드는 아래 출석부 형태의 텍스트 파일을 입력 받아 랜덤하게 PT 순서를 정하는 프로그램의 일부이다. 이를 참고하여 물음에 답하시오. (8번~9번 문제)

출석부.txt

1	201430991	김철수
2	201430991	이철수
3	201430991	박철수
4	201430991	강철수
5	201430991	한철수
...		

```

class Element{
    Element( String strInIndex, String strInName, String strInNumber )
    {
        nIndex = Integer.parseInt(strInIndex);
        strName = strInName;
        nNumber = Integer.parseInt(strInNumber);
    }
    Element( Element ele )
    {
        nIndex = ele.nIndex ; strName = ele.strName; nNumber = ele.nNumber;
    }

    int    nIndex ; String strName; int    nNumber;
}

class FileControl{
    public void write(Vector<Element> vt){
        // 함수 내용 생략...
    }
    public void read(Vector<Element> vt){
        BufferedReader reader = null;
        try {
            reader = new BufferedReader(new FileReader("출석부.txt"));
            String line = reader.readLine();

            while ( line != null ) {

                //한 줄 문자열을 탭("\t")에 의하여 구분하여 정보 분류
                ①_____...
            }
        }
    }
}

```

```

// Read next line for while condition
line = reader.readLine();

    }

    } catch (IOException ioe) {
        System.out.println(ioe.getMessage());
    } finally {
        try { if (reader!=null) reader.close(); } catch (Exception e) {}
    }
}

public class PTOOrder{

    public static void main(String[] args) {
        FileControl fc = new FileControl();
        Vector<Element> vtIn = new Vector<Element>();
        Vector<Element> vtOut = new Vector<Element>();
        fc.read(vtIn);

        //랜덤한 선택에 의해 학생 정렬
        ②_____

        fc.write(vtOut);
    }
}

```

8. ①의 영역에 읽어 들인 한 줄 문자열 line에 대하여 탭("\t")에 의하여 구분하여 정보가 분류되어 입력 아큐먼트 vt에 추가 되도록 적절한 코드를 완성 하시오.(10점)

정답:

```

StringTokenizer st = new StringTokenizer(line, "\t");
while ( st.hasMoreTokens() ) {
    String strIndex = st.nextToken();
    String strNumber = st.nextToken();
    String strName = st.nextToken();
    Element ele = new Element( strIndex, strName, strNumber );
    vt.add( ele );
}

```

아무런 코드가 없으면 0점

타입, 함수 및 파라미터 등의 잘못된 사용이 발견 되면 -2점

9. ②의 영역에 랜덤하게 선택된 학생 정보가 출력 벡터 vtOut에 발표 순서대로 추가 되도록 적절한 코드를 완성하시오.(10점)

정답

```

Random rand = new Random();//Random의 인스턴트 생성
int nSize = vtIn.size();
while ( nSize != 0 ){
    int nIndex = Math.abs( rand.nextInt(nSize) );//랜덤한 인덱스 생성
    Element ele = new Element(vtIn.get(nIndex));
    vtOut.add( ele );//출력 벡터에 추가
    vtIn.remove( nIndex );//입력 벡터에서 인덱스에 해당 되는 학생 제거
}

```



```

        nSize = vtIn.size();
    }

```

아무런 코드가 없으면 0점

타입, 함수 및 파라미터 등의 잘못된 사용이 발견 되면 -2점

정답

```

package java_book;

import java.util.*;
import java.io.*;
import java.math.*;

class Element{
    Element( String strInIndex, String strInName, String strInNumber )
    {
        nIndex = Integer.parseInt(strInIndex);
        strName = strInName;
        nNumber = Integer.parseInt(strInNumber);
    }
    Element( Element ele )
    {
        nIndex = ele.nIndex ;
        strName = ele.strName;
        nNumber = ele.nNumber;
    }

    int    nIndex ;
    String strName;
    int    nNumber;
}

class FileControl{
    public void write(Vector<Element> vt){
        try {
            FileWriter fw = new FileWriter("pt_contents.txt");//문자열의 파일 명칭
            int nSize = vt.size();
            for( int x=0 ; x < nSize ; x++ )
                fw.write( x + "\t" + vt.get(x).nNumber + "\t" +
vt.get(x).strName + "\r\n");//문자열 출력
            fw.close();
        } catch (IOException ioe) {
            System.out.println(ioe.getMessage());
        } finally {
        }
    }
    public void read(Vector<Element> vt){
        BufferedReader reader = null;
        try {
            reader = new BufferedReader(new FileReader("input.txt"));
            String line = reader.readLine();

            while ( line != null ) { // Print read line

                //한줄 문자열을 탭("\t")에 의하여 구분하여 정보 분류
                StringTokenizer st = new StringTokenizer(line, "\t");//StringTokenizer의
인스턴트 생성

                while ( st.hasMoreTokens() ) {
                    String strIndex = st.nextToken();
                    String strNumber = st.nextToken();
                    String strName = st.nextToken();
                    System.out.println(strIndex + "\t" + strNumber + "\t" +

```


정답 : 원래 의도는 **1**이 출력 되어야 할 것이다.(5점) 그러나 float의 소수점 연산 범위를 벗어나는 소수점 열 번째 자리 이후의 숫자인 0.00000000005050505050505050505051...은 float 연산에서 **버려지게 되어** fValue에는 0.005050505이 설정된다. 따라서 버려지는 숫자의 오차 누적에 의하여 1에서 약간 모자라는 숫자를 가지게 된다.(5점)

JAVA 프로그래밍 – 학기말고사 “참고”

학년: 학번: 이름: .

<http://docs.oracle.com/javase/7/docs/api/java/>의 문서를 참고하였습니다.

java.net

Class Socket

Constructors :	
Socket()	Creates an unconnected socket, with the system-default type of SocketImpl.
Socket(InetAddress address, int port)	Creates a stream socket and connects it to the specified port number at the specified IP address.
Socket(InetAddress address, int port, InetAddress localAddr, int localPort)	Creates a socket and connects it to the specified remote address on the specified remote port.
Socket(Proxy proxy)	Creates an unconnected socket, specifying the type of proxy, if any, that should be used regardless of any other settings.
Socket(SocketImpl impl)	Creates an unconnected Socket with a user-specified SocketImpl.
Socket(String host, int port)	Creates a stream socket and connects it to the specified port number on the named host.
Socket(String host, int port, boolean stream)	Deprecated. Use DatagramSocket instead for UDP transport.
Methods :	
void close()	Closes this socket.
OutputStream getOutputStream()	Returns an output stream for this socket.
InputStream getInputStream()	Returns an input stream for this socket.

java.io

Class OutputStream

Constructors :	
OutputStream()	
Methods :	
void close()	Closes this output stream and releases any system resources associated with this stream.
void flush()	Flushes this output stream and forces any buffered output bytes to be written out.
void write(byte[] b)	Writes b.length bytes from the specified byte array to this output stream.
void write(byte[] b, int off, int len)	Writes len bytes from the specified byte array starting at offset off to this output stream.
abstract void write(int b)	Writes the specified byte to this output stream.

java.io

Class DataOutputStream

Constructors :	
DataOutputStream(OutputStream out)	Creates a new data output stream to write data to the specified underlying output stream.
Methods :	
void flush()	Flushes this data output stream.
int size()	Returns the current value of the counter written, the number of bytes written

	to this data output stream so far.
void write(byte[] b, int off, int len)	Writes len bytes from the specified byte array starting at offset off to the underlying output stream.
void write(int b)	Writes the specified byte (the low eight bits of the argument b) to the underlying output stream.
void writeBoolean(boolean v)	Writes a boolean to the underlying output stream as a 1-byte value.
void writeByte(int v)	Writes out a byte to the underlying output stream as a 1-byte value.
void writeBytes(String s)	Writes out the string to the underlying output stream as a sequence of bytes.
void writeChar(int v)	Writes a char to the underlying output stream as a 2-byte value, high byte first.
void writeChars(String s)	Writes a string to the underlying output stream as a sequence of characters.
void writeInt(int v)	Writes an int to the underlying output stream as four bytes, high byte first.

java.io

Class DataInputStream

Constructors :	
DataInputStream(InputStream in)	Creates a DataInputStream that uses the specified underlying InputStream.
Methods :	
int read(byte[] b)	Reads some number of bytes from the contained input stream and stores them into the buffer array b.
int read(byte[] b, int off, int len)	Reads up to len bytes of data from the contained input stream into an array of bytes.
boolean readBoolean()	See the general contract of the readBoolean method of DataInput.
byte readByte()	See the general contract of the readByte method of DataInput.
char readChar()	See the general contract of the readChar method of DataInput.
double readDouble()	See the general contract of the readDouble method of DataInput.
float readFloat()	See the general contract of the readFloat method of DataInput.

java.util

Class StringTokenizer

Constructors :	
StringTokenizer(String str)	Constructs a string tokenizer for the specified string.
StringTokenizer(String str, String delim)	Constructs a string tokenizer for the specified string.
Methods :	
String nextToken()	Returns the next token from this string tokenizer.
boolean hasMoreTokens()	Tests if there are more tokens available from this tokenizer's string.

java.util

Class Random

Constructors :	
Random()	Creates a new random number generator.
Methods :	
int nextInt(int n)	Returns a pseudorandom, uniformly distributed int value between 0 (inclusive) and the specified value (exclusive), drawn from this random number generator's sequence.

java.util

Class Vector<E>

Constructors :		
Vector()		Constructs an empty vector so that its internal data array has size 10 and its standard capacity increment is zero.
Methods :		
boolean	add(E e)	Appends the specified element to the end of this Vector.
E	get(int index)	Returns the element at the specified position in this Vector.
E	remove(int index)	Removes the element at the specified position in this Vector.