[02]\_정태이\_20135174\_11주차\_워드정리

●지난주 실습 (갱신,검색,삽입,삭제)

●●코드

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** java.applet.\*;

**import** java.sql.\*;

**public** **class** exam7 **extends** ~~Applet~~ **implements** ActionListener {

Choice search; // 선택 박스 선언

Choice select; // 선택 박스1 선언

Label label; // 라벨 선언

TextField textField; // 값을 입력받을 텍스트 필드 선언

TextArea Area; // 결과 값을 출력할 텍스트에리어 선언

Button button; // 검색버튼 선언

**private** String url = "jdbc:inetdae7://210.115.229.77:2433";

**private** String user = "20135174";

**private** String pass = "req\*u8534as";

**public** **void** init() {

**try** {

Class.*forName*("com.inet.tds.TdsDriver");

} **catch** (ClassNotFoundException e) {

System.***out***.println("Class Loading Failed");

}

}

**public** **void** start() {

setLayout(**new** FlowLayout()); // 레이아웃 배치자

label = **new** Label("검색 조건");

add(label);

//추가

select = **new** Choice();

select.add("삽입");

select.add("삭제");

select.add("갱신");

select.add("검색");

add(select);

search = **new** Choice();

search.add("전체");

search.add("이름");

search.add("나이");

search.add("성별");

add(search);

textField = **new** TextField(10);

add(textField);

button = **new** Button("search");

add(button);

Area = **new** TextArea(10, 50);

add(Area);

button.addActionListener(**this**);

}

**public** **void** actionPerformed(ActionEvent ae) {

Connection con = **null**;

Statement stmt = **null**;

String st = ae.getActionCommand(); // 검색버튼의 string 값을 전달 받음

String item\_select = select.getSelectedItem(); // 삽입,삭제,갱신,검색 중

String item = search.getSelectedItem(); // 전체,이름,나이,성별 중

// 선택박스에서 선택된 값을 전달 받음

**if**(item\_select.equals("삽입"))

{

String n = textField.getText();

**if** (item.equals("전체")) // 선택박스에서 '삽입'+‘전체’ 가 선택 됐을 경우

{

String[] temp = n.split(",");

Insert(temp[0], Integer.*parseInt*(temp[1]) , temp[2]); // 삽입 메소드 호출

}

}

**if**(item\_select.equals("삭제"))

{

String n = textField.getText();

**if** (item.equals("전체")) // 선택박스에서 '삭제'+‘전체’ 가 선택 됐을 경우

{

customer\_Delete(n); // 삭제 메소드 호출

}

}

**if**(item\_select.equals("갱신"))

{

String n = textField.getText();

**if** (item.equals("전체")) // 선택박스에서 '삭제'+‘전체’ 가 선택 됐을 경우

{

update\_Age(); // 갱신 메소드 호출

System.***out***.println("Update Completed.");

}

}

**if**(item\_select.equals("검색"))

{

String n = textField.getText();

**if** (item.equals("전체")) // 선택박스에서 ‘전체’ 가 선택 됐을 경우

{

TotalgetDBSearch(); // 전체 검색 메소드를 호출

}

}

}

**private** **void** TotalgetDBSearch() {

Connection con = **null**;

Statement stmt = **null**;

**try** {

con = DriverManager.*getConnection*(url,user,pass);

con.setCatalog("20135174");

stmt = con.createStatement();

ResultSet result = stmt.executeQuery("SELECT \* FROM customer");

// 여기부터 아래 세 번째 까지는 검색 버튼이 클릭될 때마다 결과 창에 새로운 검색결과를 보여주기 위하여 이전에 검색한 결과를 지우기 위한 작업이다.

String count = Area.getText();

**int** c = count.length();

Area.~~replaceText~~(" ",0,c);

**while**(result.next())

{

String Name = result.getString(1);

String age = Integer.*toString*(result.getInt(2));

String sex = result.getString(3);

String value = "이름 : " +Name+ " 나이 : " +age+ " 성별 : " +sex+ "\n" ;

**int** index = Area.getText().length();

Area.~~insertText~~(value,index); // 결과창에 결과값 추가

}

con.close();

stmt.close();

}**catch**(Exception ee) {System.***out***.println(ee);}

} // try-end

// end

**private** **void** NamegetDBSearch(String n) // 이름 검색

// 사용자가 입력한 이름을 값으로 받는다.

{

Connection con = **null**;

Statement stmt = **null**;

**try**

{

con = DriverManager.*getConnection*(url,user,pass);

con.setCatalog("20135174");

stmt = con.createStatement();

ResultSet result = stmt.executeQuery("SELECT \* FROM customer WHERE name ='\"+n+\"'\");\r\n"+"'");

}

**catch**(SQLException se)

{

System.***err***.println(se.getMessage());

}

}

**private** **void** Insert(String name, **int** age, String sex)

{

**try**

{

Connection con;

Statement stmt;

// Connection 객체를 사용하여 DB에 연결을 설정

con = DriverManager.*getConnection*(url,user,pass);

con.setCatalog("20135174");

//질의문을 작성하기 위하여 Statement 객체를 생성

stmt = con.createStatement();

// 생성된 테이블에 값을 삽입

stmt.executeUpdate(

"INSERT INTO customer VALUES(' "+name+" ', ' "+age+" ',' "+sex+" ')");

// Statement,Connection 객체를 닫음

stmt.close();

con.close();

}**catch**(SQLException se){

System.***err***.println(se.getMessage());

}

}

**void** customer\_Delete(String r) {

**try**{

Connection con;

Statement stmt;

con = DriverManager.*getConnection*(url,user,pass);

con.setCatalog("20135174");

// 연결된 DB에서 질의문을 작성하기 위하여 Statement 객체를 생성

stmt = con.createStatement();

stmt.executeUpdate("delete from customer where name = ' "+r+" ' ");

stmt.close();

con.close();

}**catch**(SQLException se){

System.***err***.println(se.getMessage());

}

}

**void** update\_Age()

{

**try**{

Connection con;

Statement stmt;

con = DriverManager.*getConnection*(url,user,pass);

con.setCatalog("20135174");

// 연결된 DB에서 질의문을 작성하기 위하여 Statement 객체를 생성

stmt = con.createStatement();

stmt.executeUpdate("UPDATE customer set age = age+1 where age is not null ");

// age열이 null이 아니면 1씩 증가시킨다. -> 한살씩 업데이트

stmt.close();

con.close();

}**catch**(SQLException se){

System.***err***.println(se.getMessage());

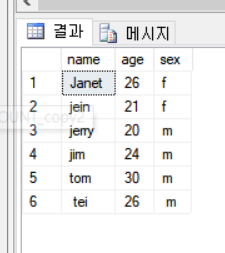
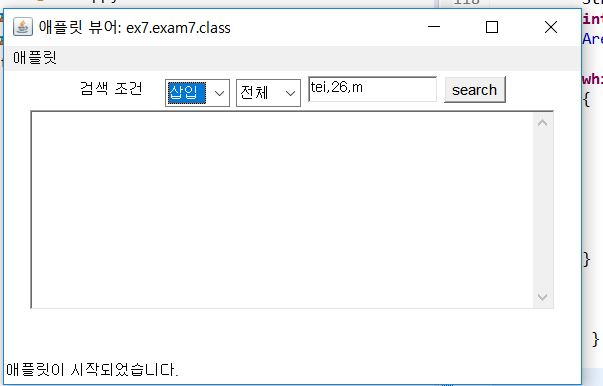
}

}

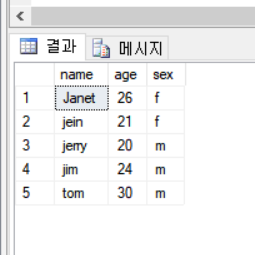
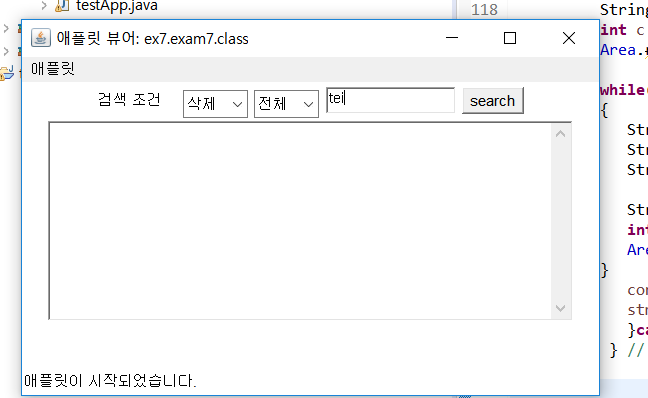
}

●결과화면

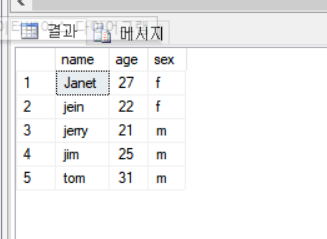
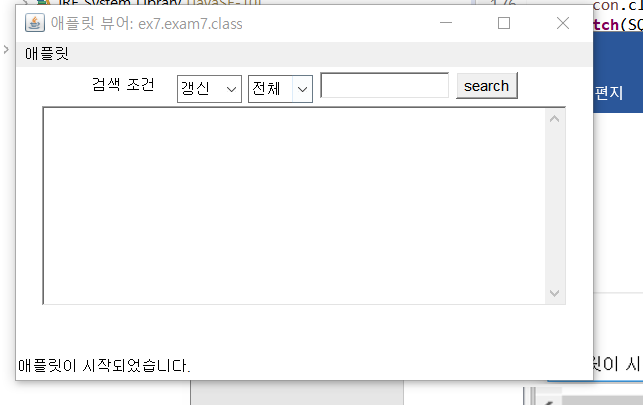
●●삽입



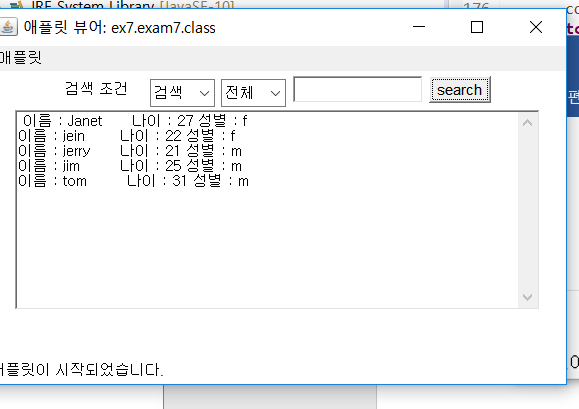
●●삭제



●●갱신



●●검색 전체



●실습1 & 실습2

●코드

**package** ex7;

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** java.applet.\*;

**import** java.sql.\*;

**public** **class** testApp **extends** ~~Applet~~ **implements** ActionListener {

Label label; // 라벨 선언

Label label2; // 라벨 선언

Label label3; // 라벨 선언

Label label4; // 라벨 선언

Label label5; // 라벨 선언

Label label6; // 라벨 선언

Label label7; // 검색

TextField tfName; // 이름

TextField tfAge; // 나이

TextField tfID; // ID

TextField tfPW; //Password

TextField tfEmail; // 이메일

TextField tfPhone; //핸드폰번호

TextArea Area; // 결과 값을 출력할 텍스트에리어 선언

Button button; // 입력버튼

Button button1; // search 버튼

**private** String url = "jdbc:inetdae7://210.115.229.77:2433";

**private** String user = "20135174";

**private** String pass = "req\*u8534as";

**public** **void** init() {

**try** {

Class.*forName*("com.inet.tds.TdsDriver");

} **catch** (ClassNotFoundException e) {

System.***out***.println("Class Loading Failed");

}

}

**public** **void** start() {

setLayout(**new** FlowLayout()); // 레이아웃 배치자

//이름

label = **new** Label("이름");

add(label);

tfName = **new** TextField(10);

add(tfName);

//나이

label2 = **new** Label("나이");

add(label2);

tfAge = **new** TextField(10);

add(tfAge);

//ID

label3 = **new** Label("ID");

add(label3);

tfID = **new** TextField(10);

add(tfID);

//password

label4 = **new** Label("password");

add(label4);

tfPW = **new** TextField(10);

add(tfPW);

//email

label5 = **new** Label("e-mail");

add(label5);

tfEmail = **new** TextField(10);

add(tfEmail);

//phone

label6 = **new** Label("phone");

add(label6);

tfPhone = **new** TextField(10);

add(tfPhone);

button = **new** Button("입력");

add(button);

label7 = **new** Label("검색");

add(label7);

button1 = **new** Button("search");

add(button1);

Area = **new** TextArea(10, 50);

add(Area);

button.addActionListener(**new** ActionListener(){

**public** **void** actionPerformed(ActionEvent ae) {

Connection con = **null**;

Statement stmt = **null**;

String st = ae.getActionCommand(); // 검색버튼의 string 값을 전달 받음

String name = tfName.getText();

String age\_temp = tfAge.getText();

String id = tfID.getText();

String pw = tfPW.getText();

String email = tfEmail.getText();

String phone = tfPhone.getText();

**int** age=Integer.*parseInt*(age\_temp);

Insert(name,age,id,pw,email,phone);

}

});

button1.addActionListener(**new** ActionListener(){

**public** **void** actionPerformed(ActionEvent ae) {

Connection con = **null**;

Statement stmt = **null**;

String st = ae.getActionCommand(); // 검색버튼의 string 값을 전달 받음

TotalgetDBSearch(); // 전체 검색 메소드를 호출

}

});

}

**private** **void** TotalgetDBSearch() {

Connection con = **null**;

Statement stmt = **null**;

**try** {

con = DriverManager.*getConnection*(url,user,pass);

con.setCatalog("20135174");

stmt = con.createStatement();

ResultSet result = stmt.executeQuery("SELECT \* FROM hallym");

// 여기부터 아래 세 번째 까지는 검색 버튼이 클릭될 때마다 결과 창에 새로운 검색결과를 보여주기 위하여 이전에 검색한 결과를 지우기 위한 작업이다.

String count = Area.getText();

**int** c = count.length();

Area.~~replaceText~~(" ",0,c);

**while**(result.next())

{

String Name = result.getString(1);

String age = Integer.*toString*(result.getInt(2));

String id = result.getString(3);

String pw = result.getString(4);

String email = result.getString(5);

String phone = result.getString(6);

String value = "이름 : " +Name+ ", 나이 : " +age+ ", ID : " +id+", PW :" +pw+", e-mail :"+email+", phone : "+phone+"\n" ;

**int** index = Area.getText().length();

Area.~~insertText~~(value,index); // 결과창에 결과값 추가

}

con.close();

stmt.close();

}**catch**(Exception ee) {System.***out***.println(ee);}

} // try-end

// end

**private** **void** NamegetDBSearch(String n) // 이름 검색

// 사용자가 입력한 이름을 값으로 받는다.

{

Connection con = **null**;

Statement stmt = **null**;

**try**

{

con = DriverManager.*getConnection*(url,user,pass);

con.setCatalog("20135174");

stmt = con.createStatement();

ResultSet result = stmt.executeQuery("SELECT \* FROM customer WHERE name ='\"+n+\"'\");\r\n"+"'");

}

**catch**(SQLException se)

{

System.***err***.println(se.getMessage());

}

}

**private** **void** Insert(String name, **int** age, String id, String pw, String email, String phone)

{

**try**

{

Connection con;

Statement stmt;

// Connection 객체를 사용하여 DB에 연결을 설정

con = DriverManager.*getConnection*(url,user,pass);

con.setCatalog("20135174");

//질의문을 작성하기 위하여 Statement 객체를 생성

stmt = con.createStatement();

// 생성된 테이블에 값을 삽입

stmt.executeUpdate(

"INSERT INTO hallym VALUES(' "+name+" ', ' "+age+" ',' "+id+" ',' "+pw+" ',' "+email+" ',' "+phone+" ')");

// Statement,Connection 객체를 닫음

stmt.close();

con.close();

}**catch**(SQLException se){

System.***err***.println(se.getMessage());

}

}

@Override

**public** **void** actionPerformed(ActionEvent arg0) {

// **TODO** Auto-generated method stub

}

}

●결과화면

