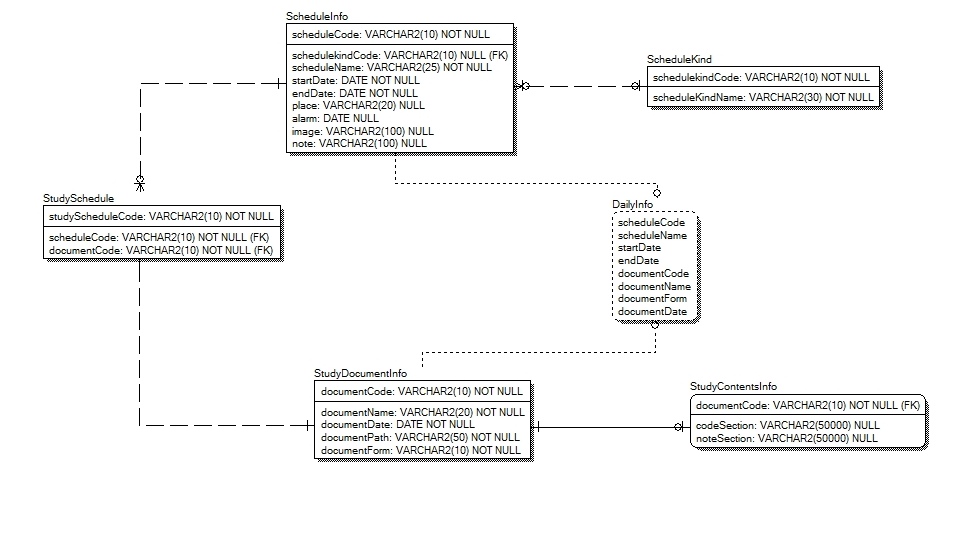
◈ Table 및 View 설계 및 생성 (Strike 팀\_팀장:장재영)



create table StudyDocumentInfo (

documentCode varchar2(10) constraint doc\_code\_pk primary key,

documentName varchar2(20) constraint doc\_name\_nn not null,

documentDate date constraint doc\_date\_nn not null,

documentPath varchar2(50) constraint doc\_path\_nn not null,

documentForm varchar2(10) constraint doc\_form\_nn not null

);

create table StudyContentsInfo (

documentCode varchar2(10) constraint doc\_code\_fk references StudyDocumentInfo(documentCode),

codeSection varchar2(4000),

noteSection varchar2(4000),

constraint doc\_con\_pk primary key(documentCode)

);

create table ScheduleKind (

scheduleKindCode varchar2(10) constraint kind\_code\_pk primary key,

scheduleKindName varchar2(40) constraint kink\_name\_nn not null

);

create table ScheduleInfo (

scheduleCode varchar2(10) constraint sch\_code\_pk primary key,

scheduleKindCode varchar2(10) constraint sch\_kind\_fk references ScheduleKind(scheduleKindCode),

scheduleName varchar2(40) constraint sch\_name\_nn not null,

startDate date constraint sch\_start\_nn not null,

endDate date constraint sch\_end\_nn not null,

place varchar2(20),

alarm date,

image varchar2(100),

note varchar2(100)

);

create table StudySchedule (

studyScheduleCode varchar2(10) constraint studySch\_code\_pk primary key,

scheduleCode varchar2(10) constraint studySch\_schCode\_fk references ScheduleInfo(scheduleCode)

constraint studySch\_schCode\_nn not null,

documentCode varchar2(10) constraint studySch\_docCode\_fk references StudyDocumentInfo(documentCode)

constraint studySch\_docCode\_nn not null,

studyScheduleDate date constraint studySch\_date\_nn not null

);

create view DailyInfoView

as

select s.scheduleCode , s.scheduleName, s.startDate, s.endDate, d.documentCode,

d.documentName, d.documentForm, d.documentDate

from ScheduleInfo s, StudyDocumentInfo d

with read only;

◈ Trigger 생성 및 단위 테스트

create or replace trigger studyDoc\_del\_tri

after delete on StudyDocumentInfo

for each row

begin

delete from StudyContentsInfo

where documentCode = :old.documentCode;

delete from StudySchedule

where documentCode=:old.documentCode;

end;

insert into StudyDocumentInfo

values ('1', '가', sysdate, 'ClassA', '.java');

insert into StudyDocumentInfo

values ('2', '나', sysdate, 'ClassB', '.java');

insert into StudyDocumentInfo

values ('3', '나', sysdate, 'ClassC', '.java');

insert into StudyContentsInfo

values ('1', 'public class ClassA', '클래스A');

insert into StudyContentsInfo

values ('2', 'public class ClassB', '클래스B');

insert into StudyContentsInfo

values ('3', 'public class ClassC', '클래스C');

insert into StudySchedule

values ('111', '2000', '1', sysdate);

insert into StudySchedule

values ('222', '2000', '2', sysdate);

insert into StudySchedule

values ('333', '2000', '3', sysdate);

select \* from StudyDocumentInfo;

select \* from StudyContentsInfo;

select \* from ScheduleInfo;

select \* from ScheduleKind;

select \* from StudySchedule;

delete from StudyDocumentInfo

where documentCode = '1';

delete from StudyDocumentInfo

where documentCode = '2';

delete from StudyDocumentInfo

where documentCode = '3';

◈ ScheduleKind, ScheduleInfo 데이터 삽입, 삭제, 수정, 검색 단위 테스트

//20개

//ScheduleKind

create table ScheduleKind(

schedulekindcode varchar2(10) primary key,

schedulekindname varchar2(30) not null

);

insert into schedulekind

values('1','이론');

insert into schedulekind

values('2','실습');

insert into schedulekind

values('3','휴식');

insert into schedulekind

values('4','운동');

insert into schedulekind

values('5','정보 수집 단계');

insert into schedulekind

values('6','유즈케이스 분석단계');

insert into schedulekind

values('7','클래스 분석단계');

insert into schedulekind

values('8','클래스 설계단계');

insert into schedulekind

values('9','클래스 상세 설계단계');

insert into schedulekind

values('10','소스코딩 및 디버깅 단계');

insert into schedulekind

values('11','문서작업 및 발표준비단계');

insert into schedulekind

values('12','최종 완료 단계');

insert into schedulekind

values('13','휴가');

insert into schedulekind

values('14','생일');

insert into schedulekind

values('15','결혼 기념일');

create table ScheduleInfo (

scheduleCode varchar2(10) constraint sch\_code\_pk primary key,

scheduleKindCode varchar2(10) constraint sch\_kind\_fk references ScheduleKind(schedulekindCode),

scheduleName varchar2(40) constraint sch\_name\_nn not null,

startDate date constraint sch\_start\_nn not null,

endDate date constraint sch\_end\_nn not null,

place varchar2(20),

alarm date,

image varchar2(100),

note varchar2(100)

);

1.

insert into scheduleinfo

values('155152001',

'3',

'공원 산책',

to\_date('2015/5/15/12/20','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/15/2/0','yyyy/mm/dd/hh24/mi'),

'율동 공원',

null,

null,

null);

2.

insert into scheduleinfo

values('155152002',

'1',

'네트워크 공부',

to\_date('2015/5/15/16/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/15/18/0','yyyy/mm/dd/hh24/mi'),

'렉토피아',

to\_date('2015/1/1/2/0','yyyy/mm/dd/hh/mi'),

null,

null);

3.

insert into scheduleinfo

values('155152003',

'5',

'벤치마킹',

to\_date('2015/5/15/19/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/15/23/0','yyyy/mm/dd/hh24/mi'),

'렉토피아',

null,

null,

null);

4.

insert into scheduleinfo

values('155162004',

'1',

'스레드',

to\_date('2015/5/16/9/0','yyyy/mm/dd/hh/mi'),

to\_date('2015/5/16/12/0','yyyy/mm/dd/hh/mi'),

'렉토피아',

null,

null,

null);

5.

insert into scheduleinfo

values('155162005',

'3',

'책읽기',

to\_date('2015/5/16/14/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/16/16/0','yyyy/mm/dd/hh24/mi'),

'우리집',

null,

null,

null);

6.

insert into scheduleinfo

values('155172006',

'4',

'헬스',

to\_date('2015/5/17/14/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/17/16/0','yyyy/mm/dd/hh24/mi'),

'스포애니',

null,

null,

null);

7.

insert into scheduleinfo

values('155172007',

'6',

'명사배치',

to\_date('2015/5/17/17/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/17/20/0','yyyy/mm/dd/hh24/mi'),

'렉토피아',

null,

null,

null);

8.

insert into scheduleinfo

values('155172008',

'6',

'명사배치',

to\_date('2015/5/17/21/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/17/22/0','yyyy/mm/dd/hh24/mi'),

'렉토피아',

null,

null,

null);

9.

insert into scheduleinfo

values('155182009',

'7',

'영문화 작업',

to\_date('2015/5/18/13/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/18/15/0','yyyy/mm/dd/hh24/mi'),

'렉토피아',

null,

null,

null);

10.

insert into scheduleinfo

values('155182010',

'12',

'중간플젝 리펙토링',

to\_date('2015/5/18/17/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/18/22/0','yyyy/mm/dd/hh24/mi'),

'렉토피아',

null,

null,

null);

11.

insert into scheduleinfo

values('155192011',

'13',

'부산 여행',

to\_date('2015/5/19/20/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/19/22/0','yyyy/mm/dd/hh24/mi'),

'부산',

to\_date('2015/5/19/7/0','yyyy/mm/dd/hh/mi'),

null,

null);

12.

insert into scheduleinfo

values('155192012',

'13',

'대구 여행',

to\_date('2015/5/19/15/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/19/22/0','yyyy/mm/dd/hh24/mi'),

'대구',

null,

null,

null);

13.

insert into scheduleinfo

values('155202013',

'13',

'유나 생일',

to\_date('2015/5/20/12/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/20/18/0','yyyy/mm/dd/hh24/mi'),

'사당역',

null,

null,

null);

14.

insert into scheduleinfo

values('155212014',

'15',

'대섭씨 결혼식',

to\_date('2015/5/21/12/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/21/17/0','yyyy/mm/dd/hh24/mi'),

'집마당',

null,

null,

'축의금 드림^-^');

15.

insert into scheduleinfo

values('155222015',

'13',

'보라카이~!',

to\_date('2015/5/22/8/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/22/18/0','yyyy/mm/dd/hh24/mi'),

'필리핀',

null,

null,

'또 가구싶당!!!');

16.

insert into scheduleinfo

values('155232016',

'14',

'교수님 생일',

to\_date('2015/5/23/8/0','yyyy/mm/dd/hh/mi'),

to\_date('2015/5/23/10/0','yyyy/mm/dd/hh/mi'),

'노원역',

null,

null,

'6월입니다');

17.

insert into scheduleinfo

values('155242017',

'8',

'DAO클래스 설계',

to\_date('2015/5/24/15/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/24/18/0','yyyy/mm/dd/hh24/mi'),

'렉토',

null,

null,

null);

18.

insert into scheduleinfo

values('155242018',

'12',

'회식',

to\_date('2015/5/24/21/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/24/23/0','yyyy/mm/dd/hh24/mi'),

'렉토',

null,

null,

null);

19.

insert into scheduleinfo

values('155252019',

'1',

'SQL복습',

to\_date('2015/5/25/12/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/25/17/0','yyyy/mm/dd/hh24/mi'),

'카페',

null,

null,

null);

20.

insert into scheduleinfo

values('155252020',

'2',

'SQL 코딩',

to\_date('2015/5/25/19/0','yyyy/mm/dd/hh24/mi'),

to\_date('2015/5/25/20/0','yyyy/mm/dd/hh24/mi'),

'카페',

null,

null,

null);

select \* from scheduleinfo order by schedulecode;

//검색

//schedule kind code로 검색

select \*

from scheduleinfo

where schedulekindcode=3;

//schedule의 코드,이름,종류이름

select schedulecode, schedulename, schedulekindname

from scheduleinfo i

join schedulekind k

on i.schedulekindcode = k.schedulekindcode;

//수정

update scheduleinfo

set schedulename='율동 공원 산책'

where schedulecode = 155152001;

//삭제

delete from scheduleinfo

where schedulecode = 155152001;

//검색(날짜별)

select \*

from scheduleinfo

where startdate between '2015/5/15' and '2015/5/17'

order by schedulecode;

select \*

from scheduleinfo

where startdate = to\_date('2015/5/15/12/20','yyyy/mm/dd/hh24/mi');

◈ StudyDocumentInfo, StudyContentsInfo 데이터 삽입, 수정, 검섹, 삭제 단위 테스트

// StudyDocumentInfo 테이블 생성.

create table StudyDocumentInfo (

documentCode varchar2(10) constraint doc\_code\_pk primary key,

documentName varchar2(20) constraint doc\_name\_nn not null,

documentDate date constraint doc\_date\_nn not null,

documentPath varchar2(50) constraint doc\_path\_nn not null,

documentForm varchar2(10) constraint doc\_form\_nn not null

);

// StudyContentsinfo 테이블 생성.

create table StudyContentsInfo (

documentCode varchar2(10) constraint doc\_code\_fk references StudyDocumentInfo(documentCode),

codeSection varchar2(4000),

noteSection varchar2(4000),

constraint doc\_con\_pk primary key(documentCode)

);

// studyDocumentInfo테이블과 studyContentInfo테이블의 모든 데이터를 읽어온다.

select \*

from StudyDocumentInfo d

Join StudyContentsInfo c

on d.documentCode=c.documentCode;

// studyDocumentInfo테이블의 2015/05/15 일의 데이터를 읽어온다.

select \* from studyDocumentInfo where documentDate=to\_date('2015/05/15','YYYY/MM/DD');

// studyDocumentInfo테이블의 2015/05/12~2015/05/15 일까지의 데이터를 읽어온다.

select \* from studyDocumentInfo where documentDate between to\_date('2015/05/12','YYYY/MM/DD') AND to\_date('2015/05/16','YYYY/MM/DD');

// studyDocumentInfo테이블의 155121001코드의 documentForm을 '.java'에서 '.txt'로 수정한다.

update studyDocumentInfo set documentForm='.txt' where documentCode='155121001';

// studyDocumentInfo테이블의 155121001코드의 행의 데이터을 삭제한다.

delete from studyDocumentInfo where documentCode='155101002';

// studyContentsInfo테이블의 15591003코드에 해당하는 데이터를 읽어온다.

select \* from studyContentsInfo where documentCode='15591003';

// studyContentsInfo테이블의 155141002코드의 codeSection의 '설 : //#1 STK 프로젝트 입니다. //#2 메인 //#3 출력합니다. 에서 'STK 클래스 입니다.'로 수정한다.

update studyContentInfo set codeSection='STK 클래스 입니다.';

// studyContentsInfo테이블의 155141002코드의 행의 데이터를 삭제한다.

delete from studyContentsInfo where documentCode='155141002';

// 데이터를 StudyDocumentInfo 테이블에 추가한다.

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('15581001','Thread1',TO\_DATE('2015/05/08', 'YYYY/MM/DD'),'15/5/8','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('15591001','GUI1',TO\_DATE('2015/05/09', 'YYYY/MM/DD'),'15/5/9','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('15591002','GUI2',TO\_DATE('2015/05/09', 'YYYY/MM/DD'),'15/5/9','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('15591003','GUI3',TO\_DATE('2015/05/09', 'YYYY/MM/DD'),'15/5/0','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155101001','DailyTest1',TO\_DATE('2015/05/10', 'YYYY/MM/DD'),'15/5/10','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155101002','DailyTest2',TO\_DATE('2015/05/10', 'YYYY/MM/DD'),'15/5/10','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155111001','RunableTest1',TO\_DATE('2015/05/11', 'YYYY/MM/DD'),'15/5/11','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155121001','RunableTest2',TO\_DATE('2015/05/12', 'YYYY/MM/DD'),'15/5/12','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm)

values('155141002','STK\_project',TO\_DATE('2015/05/14', 'YYYY/MM/DD'),'15/5/14','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm)

values('155141003','BookInfo',TO\_DATE('2015/05/14', 'YYYY/MM/DD'),'15/5/14','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155151001','JoinTest',TO\_DATE('2015/05/15', 'YYYY/MM/DD'),'15/5/15','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155151003','BookInfo',TO\_DATE('2015/05/15', 'YYYY/MM/DD'),'15/5/15','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155151004','GoodsInfo',TO\_DATE('2015/05/15', 'YYYY/MM/DD'),'15/5/15','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155151005','STKprojectOne',TO\_DATE('2015/05/15', 'YYYY/MM/DD'),'15/5/15','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155151007','STKprojectThree',TO\_DATE('2015/05/15', 'YYYY/MM/DD'),'15/5/15','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155161001','Account',TO\_DATE('2015/05/16', 'YYYY/MM/DD'),'15/5/16','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155161002','Calculator',TO\_DATE('2015/05/16', 'YYYY/MM/DD'),'15/5/16','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155161003','SharedArea',TO\_DATE('2015/05/16', 'YYYY/MM/DD'),'15/5/16','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155171001','ClientInfo',TO\_DATE('2015/05/17', 'YYYY/MM/DD'),'15/5/17','.java');

insert into StudyDocumentInfo(documentCode,documentName,documentDate,documentPath,documentForm) values('155181001','Numbers',TO\_DATE('2015/05/18', 'YYYY/MM/DD'),'15/5/18','.java');

// 데이터를 StudyContestInfo테이블에 추가한다.

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('15581001','public class Thread1(){}','Thread 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('15591001','public class GUI1(){}','GUI1 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('15591002','public class GUI2(){}','GUI2 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('15591003','public class GUI3(){}','GUI3클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155101001','public class DailyTest1(){}','DailyTest1 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155101002','public class DailyTest2(){}','DailyTest2 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155111001','public class RunableTest1(){}','RunableTest1 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155121001','

public class RunableTest2 {

public RunableTest2()

{

System.out.println( "RunableTest2 클래스 입니다." );

}

public static void main(String[] args){

new RunableTest2();

}','RunableTest2클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155141002','class STK\_project #1

{

public static void main( String[] args ) #2

{

System.out.println("Hello World!"); #3

}

}',' 설명 : //#1 STK 프로젝트 입니다.

//#2 메인

//#3 출력합니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155141003','import java.io.\*;

public class BookInfo extends GoodsInfo implements Serializable

{

private String writer;

private int page;

public BookInfo()

{

super(null, null, 0);

}

public BookInfo(String code, String name, int price, String writer, int page)

{

super(code, name, price);

this.writer = writer;

this.page = page;

}

public void setWriter(String writer)

{

this.writer = writer;

}

public void setPage(int page)

{

this.page = page;

}

public String getWriter()

{

return this.writer;

}

public int getPage()

{

return this.page;

}

public String toString()

{

String str = super.toString() + writer + " / p" + page;

return str;

}

private void writeObject(ObjectOutputStream out) throws IOException

{

out.defaultWriteObject();

}

private void readObject(ObjectInputStream in) throws IOException, ClassNotFoundException

{

in.defaultReadObject();

}

}','GoodInfo 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155151001','public class JoinTest(){}','JoinTest 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155151003','import java.io.\*;

public class BookInfo extends GoodsInfo implements Serializable

{

private String writer;

private int page;

public BookInfo()

{

super(null, null, 0);

}

public BookInfo(String code, String name, int price, String writer, int page)

{

super(code, name, price);

this.writer = writer;

this.page = page;

}

public void setWriter(String writer)

{

this.writer = writer;

}

public void setPage(int page)

{

this.page = page;

}

public String getWriter()

{

return this.writer;

}

public int getPage()

{

return this.page;

}

public String toString()

{

String str = super.toString() + writer + " / p" + page;

return str;

}

private void writeObject(ObjectOutputStream out) throws IOException

{

out.defaultWriteObject();

}

private void readObject(ObjectInputStream in) throws IOException, ClassNotFoundException

{

in.defaultReadObject();

}

}','BookInfo 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155151004','import java.io.\*;

public class GoodsInfo implements Serializable

{

private String code;

transient private String name;

private int price;

public GoodsInfo()

{

this.code = "";

this.name = "";

this.price = 0;

}

public GoodsInfo(String code, String name , int price)

{

this.code = code;

this.name = name;

this.price = price;

}

public String getCode()

{

return this.code;

}

public String getName()

{

return this.name;

}

public int getPrice()

{

return this.price;

}

public void setCode()

{

this.code = code;

}

public void setName()

{

this.name = name;

}

public void setPrice()

{

this.price = price;

}

public String toString()

{

String str = code + " / " + name + " / " + price + "원 / " ;

return str;

}

}','GoodsInfo 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155151005','class STKprojectOne #1

{

public static void main( String[] args ) #2

{

System.out.println("Hello World!"); #3

}

}','STKprojectOne 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155151007',' 코드 : public class STKprojectThree

{

public static void main(String[] args)

{

System.out.println("Hello World!");

}

}','STKprojectThree 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155161001',' 코드 : public class Account

{

private String accountNo;

private String ownerName;

private int balance;

public Account()

{

this.accountNo = "";

this.ownerName = "";

this.balance = 0;

}

public Account(String accountNo, String ownerName, int balance)

{

this.accountNo = accountNo;

this.ownerName = ownerName;

this.balance = balance;

}

public void setAccountNo(String accountNo)

{

this.accountNo = accountNo;

}

public void setOwnerName(String ownerName)

{

this.ownerName = ownerName;

}

public void setBalance(int balance)

{

this.balance = balance;

}

public String getAccountNo()

{

return this.accountNo;

}

public String getOwnerName()

{

return this.ownerName;

}

public int getBalance()

{

return this.balance;

}

public int deposit(int balance)

{

System.out.println(" ## 입금합니다~~ ");

this.balance += balance;

return this.balance;

}

public int withdraw(int amount) throws Exception

{

if(this.balance < amount)

{

throw new Exception(" 잔액이 부족합니다. ");

}

System.out.println(" ## 출금합니다~~ ");

this.balance -= amount;

return amount;

}

public String toString() // toString() 오버라이드

{

String str = "계좌 : " + accountNo +

"\n예금주 이름 : " + ownerName +

"\n잔액 : " + balance + "\n";

return str;

}

}','Account 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155161002','

import javax.swing.\*;

import java.awt.event.\*;

import java.awt.\*;

public class Calculator extends JFrame implements ActionListener, KeyListener{

final int MAX\_INPUT\_LENGTH = 20; //최대한 입력 가능한 길이를 제한하고

//각 모드별로 index를 부여

final int INPUT\_MODE = 0;

final int RESULT\_MODE = 1;

final int ERROR\_MODE = 2;

int displayMode;

boolean clearOnNextDigit; //화면에 표시될 숫자를 지울지 말지 결정하는 녀석

double lastNumber; //마지막에 기억될 수

String lastOperator; // 마지막에 누른 연산자를 기억.

private JLabel output; //숫자가 표시될 공간

private JButton buttons[]; //각 버튼을 배열로 생성할 것임.

private JPanel masterPanel, btnPanel, ctrPanel; //버튼과 레이블을 배치할 공간

//GUI를 생성해보자.

public Calculator(){

super("STK\_계산기");

setBackground(Color.gray); //프레임의 배경은 회색

masterPanel = new JPanel();

output = new JLabel("0", JLabel.RIGHT); // 레이블의 초기 값은 0

output.setBackground(Color.WHITE);

output.setOpaque(true);

buttons = new JButton[23];

btnPanel = new JPanel();

//숫자 버튼 만듦0에서9까지.

for (int i=0 ; i<=9 ; i++){

buttons[i] = new JButton(String.valueOf(i));

}

//연산 버튼 만듦.

buttons[10] = new JButton("±");

buttons[11] = new JButton(".");

buttons[12] = new JButton("=");

buttons[13] = new JButton("/");

buttons[14] = new JButton("\*");

buttons[15] = new JButton("-");

buttons[16] = new JButton("+");

buttons[17] = new JButton("√");

buttons[18] = new JButton("1/x");

buttons[19] = new JButton("%");

//컨트롤 버튼 만듦.

buttons[20] = new JButton("←");

buttons[21] = new JButton("CE");

buttons[22] = new JButton("C");

//각 버튼의 색을 다르게

for(int i=0 ; i<buttons.length ; i++){

if(i<10){

buttons[i].setForeground(Color.black); //숫자 버튼들

}else if(i>=10 && i<21){

buttons[i].setForeground(Color.blue); //연산자 버튼들

}else{

buttons[i].setForeground(Color.red); //컨트롤 버튼들.

}

}

//패널에 숫자버튼 및 연산자 버튼 배치

btnPanel.setLayout(new GridLayout(4,5,2,2));

for(int i=7; i<=9 ; i++){

btnPanel.add(buttons[i]);

}

btnPanel.add(buttons[13]); // /

btnPanel.add(buttons[17]); // √

for(int i=4; i<=6 ; i++){

btnPanel.add(buttons[i]);

}

btnPanel.add(buttons[14]); // \*

btnPanel.add(buttons[18]); // 1/x

for(int i=1; i<=3 ; i++){

btnPanel.add(buttons[i]);

}

}','Calculate 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155161003','public class SharedArea

{

Account account1;

Account account2;

synchronized void transfer(int amount)

{

try

{

account1.withdraw(100000);

System.out.println(" 몽룡계좌 : 10만원 인출, ");

account2.deposit(100000);

System.out.println(" 춘향계좌 : 10만원 입금");

}

catch (Exception e)

{

e.printStackTrace();

}

}

synchronized int getTotal()

{

return account1.getBalance() + account2.getBalance();

}

}','SharedArea 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155171001','public class ClientInfo

{

private String name;

private String id;

private String password;

private String phoneNo;

private String address;

public ClientInfo()

{

this.name="";

this.id="";

this.password="";

this.phoneNo="";

this.address="";

}

public ClientInfo(String name,String id,String password)

{

this.name=name;

this.id=id;

this.password=password;

}

public ClientInfo(String name,String id,String password,String phoneNo,String address)

{

this(name,id,password);

this.phoneNo=phoneNo;

this.address=address;

}

public void changePassword(String Password)

{

this.password=password;

}

public String getName()

{

return this.name;

}

public String getId()

{

return this.id;

}

public String getPassword()

{

return this.password;

}

public String getPhoneNo()

{

return this.phoneNo;

}

public String getAddress()

{

return this.address;

}

public void setName(String name)

{

this.name=name;

}

public void setId(String id)

{

this.id=id;

}

public void setPassword(String password)

{

this.password=password;

}

public void setPhoneNo(String phoneNo)

{

this.phoneNo=phoneNo;

}

public void setAddress(String address)

{

this.address=address;

}

}

','ClientInfo 클래스 입니다.');

insert into StudyContentsInfo(documentCode,codeSection,noteSection) values('155181001','public class Numbers

{

private int[] num;

private int total;

private int average;

public Numbers()

{

this.num=null;

}

public Numbers(int[] num)

{

this.num=num;

}

public int getTotal()

{

for(int cnt=0;cnt<this.num.length;cnt++)

{

this.total+=num[cnt];

}

return this.total;

}

public int getAverage()

{

return this.average=this.total/this.num.length;

}

public int[] getNum()

{

return this.num;

}

public void setNum(int[] num)

{

this.num=num;

}

}

','Numbers 클래스 입니다.');