My Ka Kao Talk

Go to Github

Java Swing과 MultiThread Socket, JDBC(MySQL) 을 이용하여 카카오톡(구버전) 을 구현해 보았습니다.

기능

- 회원 등록
- 친구 검색
- 친구 추가/삭제
- 프로필 수정
- 채팅방 개설
- 채팅방 참가/나가기
- 채팅방 대화 친구 추가

설명

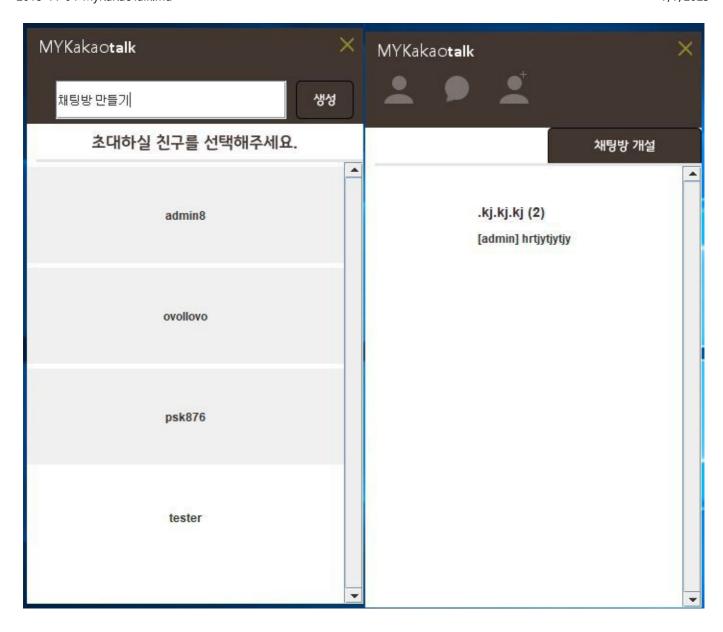
좌측) 로그인 화면 우측) 회원가입 화면



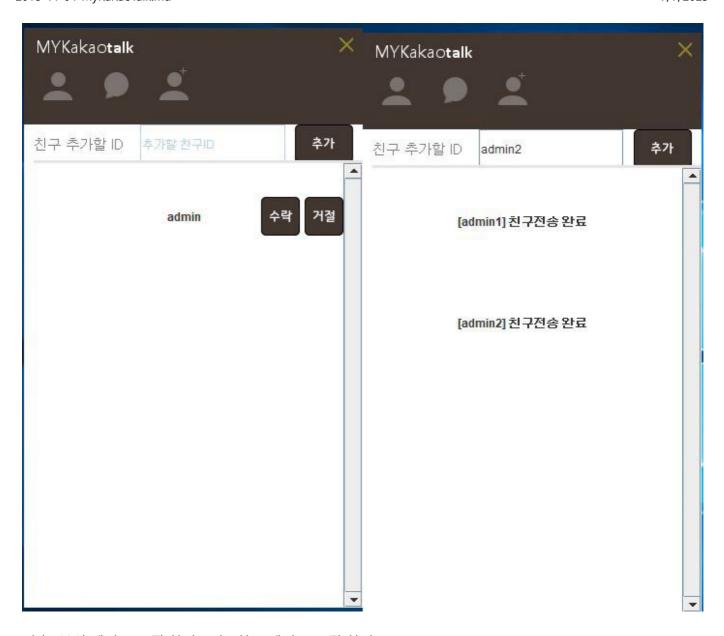
메인화면



좌측) 채팅방 설정 우측) 채팅방 리스트



친구 추가 좌측) admin1 계정 우측) admin 계정



좌측) 본인 계정 프로필 화면 우측) 친구 계정 프로필 화면



DataBase

MyKaKao DB Table

chattingroom	
PK roomID	
PK roomName	
PK member	

singlechat	
PK senderID PK roomID chatContent	
	PK senderID PK roomID

chatlog	
PK senderID	
PK roomID	
chatContent	
PK chatTime	

user

PK userID

userPassword

userName

userEmail

userTel

prifile

addfriendlist

PK senderID FK(user userID)

PK receiverID FK(user userID)

friend

PK userID FK(user userID)

PK friendID FK(user userID)

• addfriendlist

Field	Type	Null	Key	Default	Extra
senderID	varchar(24)	NO	PRI	NULL	
receiverID	varchar(24)	NO	PRI	NULL	_

• chatlog

Field	Туре	Null	Key	Default	Extra
senderID	varchar(24)	NO	PRI	NULL	
roomID	int(11)	NO	PRI	NULL	
chatContent	varchar(100)	YES		NULL	
chatTime	datetime	NO	PRI	00:00:00:00:00	

• chattingroom

Field	Туре	Null	Key	Default	Extra
roomID	int(11)	NO	PRI	NULL	
roomName	varchar(48)	NO	PRI	NULL	
member	varchar(24)	NO	PRI	NULL	

• friend

	Field	Type	Null	Key	Default	Extra
	userID	varchar(24)	NO	PRI	NULL	
ų.	friendID	varchar(24)	NO	PRI	NULL	

• singlechat

Field	Туре	Null	Key	Default	Extra
senderID	varchar(24)	NO	PRI	NULL	
roomID	int(11)	NO	PRI	NULL	
chatContent	varchar(100)	YES		NULL	
chatTime	datetime	NO	PRI	00:00:00:00:00	

user

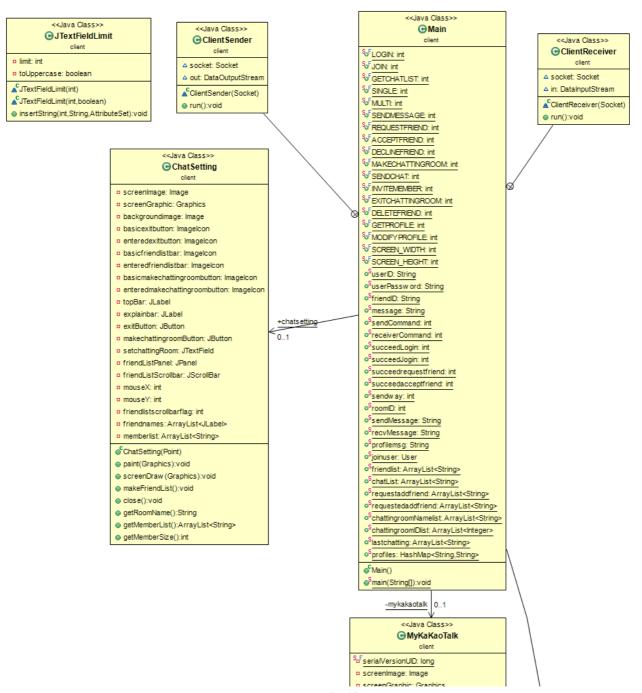
Field	Туре	Null	Key	Default	Extra
userID	varchar(24)	NO	PRI	NULL	

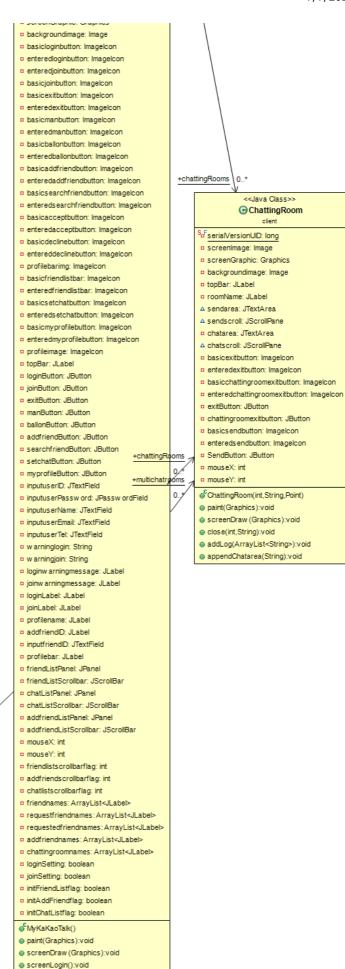
Field	Туре	Null	Key	Default	Extra
userPassword	varchar(24)		PRI	NULL	
userName	varchar(16)	NO		NULL	
userEmail	varchar(32)	NO		NULL	
userTel	varchar(16)	NO		NULL	
profile	varchar(32)	YES			

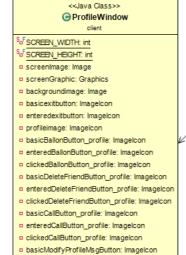
Class Diagram

Client

client







enteredModifyProfileMsgButton: Imagelcon

□ topBar: JLabel

mouseX: int

mouseY: int

userIDLabel: JLabel

exitButton: JButton

ballonButtonProfile: JButton

a callButtonProfile: JButton

a deleteFriendButtonProfile: JButton

modifyProfileMsgButton: JButton

- profilew indows

- profilew in

changeFriendList():void

changeChattingList():void

changeAddFriend():void

makeFriendList():void

makeChatList():void

profileLabel: JLabel

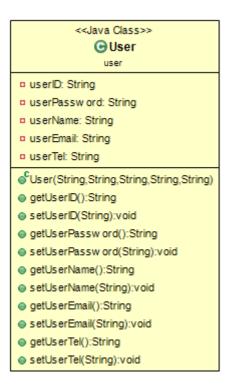
© ProfileWindow (Point,String,String)

paint(Graphics):void

screenDraw (Graphics):void

close(String):void

user



Server

server

makeAddFriendList():void

makeFriendNameList(int):void

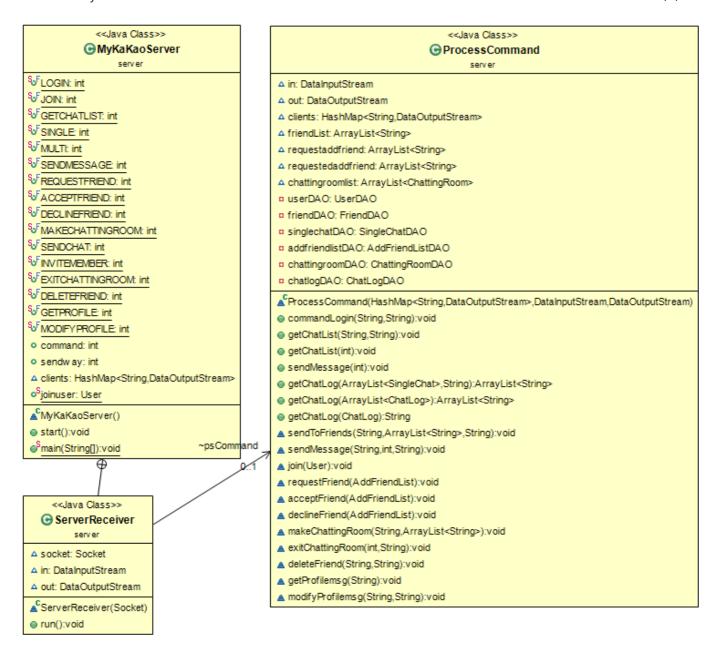
makeChattingRoomList(int):void
 remakeAddFriendList():void
 makeRequestFriendList(int):void
 makeRequestedFriendList(int,int):void
 commandLogin(int):void
 commandJoin(int):void

commandSendMessage(String,String):void
 commandSendChat(int,String):void

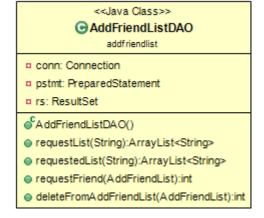
remakeFriendList():void

remakeChatList():void

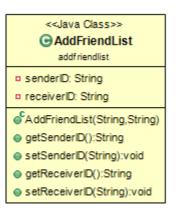
getLoacation():Point
 openChattingRoom(int, String):void
 openChattingRoom(String):void
 openProfileWindow (String):void
 reopenProfileWindow (String):void
 closeProfileWindow (String):void



addfriendlist



chatlog



<<Java Class>> ChatLogDAO

chatlog

- a conn: Connection
- pstmt: PreparedStatement
- nrs: ResultSet
- ChatLogDAO()
- getChatList(int):ArrayList<ChatLog>
- getLastChat(int):ChatLog
- storeMessage(ChatLog):int

<<Java Class>>

© ChatLog chatlog

- △ senderID: String
- △ roomID: int
- △ chatContent: String
- △ chatTime: String
- ChatLog()
- getSenderID():String
- setSenderID(String):void
- getRoomID():int
- setRoomID(int):void
- getChatContent():String
- setChatContent(String):void
- getChatTime():String
- setChatTime(String):void
- setCurrentTime():void

chattingroom

<<Java Class>>

○ ChattingRoom ○

chattingroom

- △ roomID: int
- △ roomName: String
- △ member: String
- ChattingRoom(int,String)
- getRoomID():int
- setRoomID(int):void
- getRoomName():String
- setRoomName(String):void
- getMember():String
- setMember(String):void

<<Java Class>>

ChattingRoomDAO

chattingroom

- conn: Connection
- pstmt: PreparedStatement
- nrs: ResultSet
- ChattingRoomDAO()
- makeChattiingRoom(String,ArrayList<String>):int
- exitChattingRoom(int,String):int
- cleanChattingRoom(int):void
- cleanChatLog(int):int
- getMaxRoomID():int
- getChattingRoomList(String):ArrayList<ChattingRoom>
- getMemberCount(int):int
- getChattingRoomMembers(int):ArrayList<String>
- getRoomName(int):String

friend

<<Java Class>>



friend

- userID: String
- friendID: String
- Friend(String,String)
- getUserID():String
- setUserID(String):void
- getFriendID():String
- setFriendID(String):void

<<Java Class>>

FriendDAO

friend

- a conn: Connection
- pstmt: PreparedStatement
- nrs: ResultSet
- FriendDAO()
- makeFriend(String,String):int
- friendList(String):ArrayList<String>
- deleteFrined(String,String):int

singlechat

<<Java Class>> SingleChatDAO

singlechat

- a conn: Connection
- pstmt: PreparedStatement
- nrs: ResultSet
- SingleChatDAO()
- getChatList(String,String):ArrayList<SingleChat>
- sendMessage(SingleChat):int
- storeMessage(SingleChat):int

<<Java Class>>

- △ senderID: String
- △ receiverID: String
- △ chatContent: String
- △ chatTime: String
- SingleChat()
- getSenderID():String
- setSenderID(String):void
- getReceiverID():String
- setReceiverID(String):void
- getChatContent():String
- setChatContent(String):void
- getChatTime():String
- setChatTime(String):void
- setCurrentTime():void

user

<<Java Class>>

OUser

ıser

- userID: String
- userPassword: String
- userName: String
- userEmail: String
- userTel: String
- ^CUser(String,String,String,String,String)
- getUserID():String
- setUserID(String):void
- getUserPassw ord():String
- setUserPassw ord(String):void
- getUserName():String
- setUserName(String):void
- getUserEmail():String
- setUserEmail(String):void
- getUserTel():String
- setUserTel(String):void

IICA

- a conn: Connection
- pstmt: PreparedStatement
- nrs: ResultSet
- ^CUserDAO()
- login(String,String):int
- join(User):int
- updateProfile(String,String):int
- getProfilemsg(String):String