**KBO PLAY**

작성자 : 김민기

작성일 : 2017.08.23

**1. 개발목적 : JDBC를 연습하기 위한 팀프로젝트로, Oracle 11을 이용하여 DB구축하여, JAVA내에서 JDBC를 이용해 DB를 핸들링연습을 목적으로 한다.**

**2. 개발환경 : Windows7, Eclipse Java Mars.2, JAVA JDK 1.8, ORACLE XE 11**

**3. 주요기능 :**

**4. DB 구조 : “테이블구조.xlsx” 참조**

**5. 프로그램구조 :**

**6. 주요기능 상세설명**

1. **접속 인터페이스(LoginController)**

유저가 접속할 최초 화면 서버정보는 소스에 포함되어 있으며 계정생성기능과 생성된 계정을 입력하여 서버에 접속할 수 있도록 유도한다. ID와 PW 입력을 유도하며 PW는 2회 입력해 동일한 정보가 들어와야 한다. 계정생성 완료 후 선호구단을 물어보고 선호구단에 알맞은 초기선수, 포지션을 생성해줘야 한다.

계정생성에 관한 세부내용은 “테이블구조.xlsx”을 참고한다.

계정생성에 관한 코드는 기존 dolls프로젝트를 참고한다.

**--- 안녕하세요. KBO 2016리그에 오신 여러분을 환영합니다. --**

**1. 접속 -> accountdao.userLogin() -> MainController.print()**

**2. 계정생성 -> accountdao.makeUser();**

**0. 종료 -> system.exit(0);**

1. **메인화면(MainController)**

사용자가 접속시 사용자의 상태와 다음작업으로 유도해주는 화면으로 아래 정보를 출력해야 한다.

**ACCOUNT\_[USERID] 구단주님 환영합니다.**

**LV : ACCOUNT\_[USERLV], [필요경험치] ACCOUNT\_[KRW]원**

**승 : ACCOUNT\_[WIN] / 패 : ACCOUNT\_[LOSS] 승률 : [승률계산]%**

**1. 상점** -> StoreController.print()

**2. 구단관리** -> managerController.print()

**3. 경매장 -**> auctionController.print()

**4. 서버접속** -> util.connectServer();

**0. 프로그램종료** -> system.exit(0);

1. **상점(StoreController)**

사용자가 보유한 원화로 선수구매 및 판매를 도와줘야 한다.

구매 : 구매를 누를 때 마다 랜덤하게 선수가 생성되며(등급에 따라 등장확률이 다르다), 높은수준의 재화를 요구한다.

판매 : 유저가 필요없는 카드를 판매하며 낮은수준의 보상을 재공한다.

**KBO 마켓에 오신 것을 환영합니다.**

1. **투수구매 ->**

Picherinfodao.Printradomlist() -> 선수 information에서 등급을 이용하여 랜덤하게 리스트출력

보유금액이 충분한지 확인 보유금액이 충분하면

SQL Insert를 이용해 새로운 선수 추가

1. **야수구매 ->**

<위동>

1. **투수판매 ->**

Prinpitcherlist() -> 보유선수 출력후 시퀀스를 입력받음  
-> 적은 보상을 지급

SQL Delete를 이용해 선수 삭제

1. **야수판매 ->**

<위동>

1. **구단관리**

사용자가 보유한 선수를 이용해 경기에 참가할 20인엔트리 구성을 도와줘야 한다.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **투수엔트리** | | | | | | | | |
| **1선발** | **2선발** | **3선발** |  | **1불펜** | **2불펜** | **3불펜** | **4불펜** |  |
| st1 | st2 | st3 |  | pb1 | pb2 | pb3 | pb4 |  |
| **야수엔트리** | | | | | | | | |
| **포수** | **1루수** | **2루수** | **3루수** | **유격수** | **좌익수** | **중견수** | **우익수** | **지명** |
| 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | D |
| **대기1** | h1 | **대기2** | h2 | **대기3** | h3 | **대기4** | h4 |  |

1. **선발리스트 출력** -> StaringlistPrint ()

보유중인 선수출력

1. **선발투수 설정** -> setStartingPicher()

**----- 선발라인업 -----**

StartingPicherPrint() -> 선발투수 상태표시

**----- 보유투수 -----**

StartingPicherPicherPrint() -> 선발투수리스트 출력

**변경할 포지션을 선택해 주세요(1선발 : st1, 2선발 : st2, 3선발, st3)**

스캐너를 통해 포지션을 입력받음

**교체할 투수의 고유번호를 선택해 주세요.**

스캐너를 통해 시퀀스번호를 입력받음

기존 투수의 포지션값을 -> null, 입력받은 선수로 교체

1. **불펜투수 설정** -> setBullPenPicher()

**----- 선발라인업 -----**

BullPenPicherPrint() -> 불펜투수 상태표시

**----- 보유투수 -----**

BullPenPicherlist() -> 불펜투수리스트 출력

**변경할 포지션을 선택해 주세요(1구원 : pb1, 2구원 : pb2, 3구원, pb3, 4구원 pb4)**

스캐너를 통해 포지션을 입력받음

**교체할 투수의 고유번호를 선택해 주세요.**

스캐너를 통해 시퀀스번호를 입력받음

기존 투수의 포지션값을 -> null, 입력받은 선수로 교체

1. **선발 설정**