1. 로그인  
   SELECT m\_email, m\_pwd, count(\*) AS cnt FROM Member WHERE m\_email=?;  
   IF(cnt ==1){  
    // pwd 검사 후, 로그인 세션 부여  
    // pwd 가 다르면 비밀번호 다름 메시지 출력  
   }else{  
    // email 확인하라는 메시지  
   }  
     
   ★ 참고  
   m\_pwd는 64바이트로 된 암호화 문자열
2. 회원가입  
   // 회원의 기본적인 정보 DB저장 쿼리  
   INSERT INTO   
   Member(m\_email, m\_pwd, m\_sex, m\_year, m\_month, m\_day, m\_balance)  
   VALUES(?, ?, ?, ?, ?, ?, 0);  
     
   // 회원 이미지 사진 저장  
   INSERT INTO  
   MemberImg(m\_email, dp\_num, mi\_imgName) VALUES(?, 1, “default.png”);  
     
   ★ 참고  
   mi\_imgName은 사용자가 올린 이미지 파일명
3. 사용자가 입장했을 경우, **리더기로부터 c\_num(클럽고유번호) 와 m\_email(사용자이메일)을 얻어옴**// 사용자가 입장하면, History Table로 기록 저장  
   INSERT INTO History(m\_email, c\_num, h\_date, h\_dayName, h\_status) VALUES(?, ?, now(), (SELECT case DAYOFWEEK(now()) WHEN '1' THEN '일요일' WHEN '2' THEN '월' WHEN '3' THEN '화' WHEN '4' THEN '수' WHEN '5' THEN '목' WHEN '6' THEN '금' WHEN '7' THEN '토' END AS dayofweek), 1);
4. 사용자가 금액을 **충전**하는 경우, **m\_email(사용자이메일) 과 mc\_charge(충전금액)을 가져옴**  
   INSERT INTO MoneyCharge(m\_email, mc\_charge, mc\_date) VALUES(?, ?, now());  
   UPDATE Member SET m\_balance=m\_balance + ? WHERE m\_email=?;

★ 참고  
m\_balance = m\_balance + 충전금액

* 1. 사용자가 **결제**했을 경우, **m\_email(사용자이메일)과 c\_num(클럽고유번호) 그리고 pc\_code(상품고유번호)**를 넘겨줌 🡪 결제내역을 Payment 테이블로 저장  
       
     SELECT m\_balance FROM Member WHERE m\_email=?; // 잔액조회  
     SELECT cp\_price FROM ClubProduct WHERE c\_num=? and pc\_code=?; // 상품가격 조회  
     if (m\_blance – cp\_price < 0){  
      // msg “결제금액 부족”  
     }else{  
       
     INSERT INTO Payment(m\_email, pc\_code, c\_num, p\_date, p\_dayName) VALUES(?, ?, ?, now(),(SELECT case DAYOFWEEK(now()) WHEN '1' THEN '일요일' WHEN '2' THEN '월' WHEN '3' THEN '화' WHEN '4' THEN '수' WHEN '5' THEN '목' WHEN '6' THEN '금' WHEN '7' THEN '토' END AS dayofweek))  
       
     UPDATE Member SET m\_balance = m\_balance - ? WHERE m\_email=?;

1. Profile을 클릭하여, 사진첨부할 경우,  
   UPDATE MemberImg SET mi\_imgName=? WHERE m\_email=?;  
     
   ★ 참고  
   mi\_imgName에서의 ?는 “사용자이메일+이미지이름”으로 하여 저장하도록 한다
2. Main Page로 던져줄 때,  
   1) 메인페이지에서 Profile사진을 불러오기 할 때,   
   SELECT CONCAT(d.dp\_dirPath,'/',m.mi\_imgName) AS path FROM DirectoryPath d JOIN MemberImg m ON d.dp\_num=m.dp\_num WHERE m.m\_email=?;  
     
   ★ 참고  
   프로필 경로 **Path = d.dp\_dirPath + “/” + m.mi\_imgName** 을 통하여 생성한다.  
     
   2) SELECT c.c\_num, c.c\_clubName, ci.ci\_imgName FROM Club c JOIN ClubImg ci ON c.c\_num=ci.c\_num;
3. History Page에서 사용할 쿼리
   1. History 페이지에 처음 접속했을 때,  
      SELECT c.c\_clubName, h.h\_date FROM History h JOIN Club c WHERE c.c\_num=h.c\_num and h.m\_email="?" ORDER BY h\_date DESC limit 15;
   2. 입장 내역 탭 클릭시, 출력(= 15개까지만)   
      SELECT c.c\_clubName, h.h\_date FROM History h JOIN Club c WHERE c.c\_num=h.c\_num and h.m\_email="?" ORDER BY h\_date DESC limit 15;
   3. 결제내역 출력 🡪 클럽명 상품명 상품가격 날짜  
      SELECT F.c\_clubName, E.pc\_name, E.cp\_price, E.p\_date FROM (SELECT C.c\_num, D.pc\_name, C.cp\_price, C.p\_date FROM (SELECT A.c\_num, A.pc\_code, B.cp\_price, A.p\_date FROM (SELECT c\_num, pc\_code, p\_date FROM Payment WHERE m\_email=?) AS A JOIN ClubProduct B WHERE A.c\_num=B.c\_num and A.pc\_code=B.pc\_code) AS C JOIN ProductCode D WHERE C.pc\_code=D.pc\_code) AS E JOIN Club F WHERE E.c\_num=F.c\_num ORDER BY E.p\_date DESC LIMIT 15

* 1. 충전내역 출력  
     SELECT mc\_charge, mc\_date FROM MoneyCharge WHERE m\_email=? ORDER BY mc\_date DESC limit 15;

1. ClubDetail Page 쿼리
   1. 클럽디테일 페이지에 처음 접속했을 떄, 쓰일 쿼리  
      SELECT F.ci\_startTime, F.ci\_endTime, F.ci\_price, F.ci\_phone, F.ci\_infoDJ, F.ci\_guest, F.ci\_event, E.path FROM (SELECT C.c\_num, C.c\_clubName, CONCAT(D.dp\_dirPath,'/',C.ci\_imgName) AS path FROM (SELECT A.c\_num, A.c\_clubName, B.ci\_imgName, B.dp\_num FROM (SELECT c\_num, c\_clubName FROM Club WHERE c\_num=?) A JOIN ClubImg B WHERE A.c\_num=B.c\_num) C JOIN DirectoryPath D WHERE C.dp\_num=D.dp\_num) E JOIN ClubInfo F WHERE E.c\_num=F.c\_num;
   2. CLUB 탭을 선택했을 때, 쓰일 쿼리  
      SELECT \* FROM ClubInfo WHERE c\_num=?;
   3. DRINK 탭을 선택했을 때, 쓰일 쿼리 🡪 상품이미지, 상품명, 상품가격  
      SELECT C.c\_num, C.pc\_code, C.pc\_name, C.cp\_price, CONCAT(D.dp\_dirPath,'/',C.pc\_saveName) AS path FROM (SELECT A.c\_num, A.pc\_code, B.pc\_name, B.pc\_saveName, B.dp\_num, A.cp\_price FROM (SELECT \* FROM ClubProduct WHERE c\_num=?) A JOIN ProductCode B WHERE A.pc\_code=B.pc\_code) C JOIN DirectoryPath D WHERE C.dp\_num=D.dp\_num;
   4. 차트
2. DB에서 Hadoop으로 넘기는 데이터 추출 쿼리
   1. 클럽에 입장한 사람들의 데이터 추출 쿼리 🡪 성별 나이 추출  
      SELECT m\_email, m\_sex, m\_year FROM Member WHERE m\_email IN (SELECT m\_email FROM History WHERE (DATE(h\_date)=DATE(now()) or DATE(h\_date)=DATE(DATE(now())-1)) and h\_status=1);
   2. 한 클럽에서 한 달동안 판매한 상품목록 출력 쿼리  
      SELECT B.pc\_name FROM (SELECT \* FROM Payment WHERE c\_num=?) A JOIN ProductCode B WHERE A.pc\_code=B.pc\_code and MONTH(A.p\_date)=MONTH(now())
3. 사용자가 퇴장하기 버튼을 눌렀을 경우,
4. 메인에서 지역 / 테마 Ajax처리시, 사용할 쿼리
   1. 지역 클릭 시,   
      SELECT C.c\_num, C.c\_clubName, CONCAT(D.dp\_dirPath,'/',C.ci\_imgName) AS path FROM (SELECT A.c\_num, A.c\_clubName, B.dp\_num, B.ci\_imgName FROM (SELECT c\_num, c\_clubName FROM Club WHERE d\_districtCode=?) A JOIN ClubImg B WHERE A.c\_num=B.c\_num) C JOIN DirectoryPath D WHERE C.dp\_num=D.dp\_num
   2. 테마 클릭 시,   
      SELECT C.c\_num, C.c\_clubName, CONCAT(D.dp\_dirPath,'/',C.ci\_imgName) AS path FROM (SELECT A.c\_num, A.c\_clubName, B.dp\_num, B.ci\_imgName FROM (SELECT c\_num, c\_clubName FROM Club WHERE t\_themeCode=?) A JOIN ClubImg B WHERE A.c\_num=B.c\_num) C JOIN DirectoryPath D WHERE C.dp\_num=D.dp\_num