●alertmail\_agura\_v.100.sql - tb\_trn\_rentalを基準にtb\_mst\_book、tb\_mst\_userをleft joinに接続して、延滞本をすぐに出力できるように出力している。

●insertRentdataAgua.sql - 延滞された本を出力するためには、ダミーデータが必要なため、ダミーの本データを入れてくれるsqlを作成した。第一線はrollback;を入れて、以前の誤ったsqlを再戻したてcommit;を毎回入れて、データがうまく入るした。

●insertUserinfoAgura.sql - ユーザーの情報を任意に決めてデータを入れるsqlを作成team1メンバーと共有している。

●Overdue\_list.java - Overdue\_listServletで必要な情報をよく保存してjspファイルでよく送ることができるようtoString（）の中に、必要な情報を受けて来る順番に並べて入れた。最初にdueDate順序が間違って一定部分がよく出ないていたが、再配置をしてよく表現することができるようにした。

●overdue\_list.jsp - 既存のタムワースのフラットのリストUIを守りながら情報がよく出るように登録をして送信ボタンを催促状態に基づいて、ボタンがナオドンガ出ないように、最初にtaglibのif関数を使用したが、催促中でもメールを送信するできるようにボタンを常に見えるようにした。催促中の状態はtrue or falseしかなかったが、これもtaglibのif関数を使用してtrueのときは、送信完了、falseのときは、未送信が出てくることができるよう状態の変更をして、ユーザーが一度に催促メール送信状態を確認することができた。

●Overdue\_listServlet.java - DBからbook\_id、title、author、due\_date、user\_id、name、alert情報を受けてきた後、listに入れた後、returnListに入れrequest.setAttribute（「returnList"、returnList）;を介してreturnListをoverview\_list.jspにDBから受信した情報を渡してくれた。特に催促メール送信状態をすぐに適用するためにstatus1にbook\_idが入らない場合は、単にdbでの状態を受けてきてbook\_idを受けてきた場合、executeUpdateを実行して、催促メール送信状態を反映する。

●Overdue\_send.java - Overdue\_sendServelt.javaでdbの値をよく受けて来ることができるようemail、name、title、dueDate、bookIdのsetfunctionとgetfunctionを作った。

●overdue\_send.jsp - book\_idをOverdue\_sendServelt.javaで受けてきてbook\_idに関連するname、duedate、emailを受けて自動的に催促メールを作成してくれる。そこに送信を押すと、Overdue\_listServlet.javaに移動して催促メール送信状態に反映した状態で、期限切れの本のリストを出力する。

●Overdue\_sendServlet.java - book\_idをOverdue\_sendServelt.javaで受けてきてbook\_idに関連するname、duedate、emailを受けoverdue\_send.jspに送ってくれる。

●overDueSql\_agura\_ver1.01.sql - 期限切れの本のリストを表現するためにtb\_trn\_rentalを基準にtb\_mst\_book、tb\_mst\_userをleft joinに接続して、延滞本をすぐに出力できるように出力している。

●updateAlerttoFalse\_agura.sql - 催促メール送信状態がDBの両方にtrueの状態で変わるため、false状態に更新させるsql

●updateStatus\_agura.sql - 催促メール送信状態をDBに設定しているsql

●alertmail\_agura\_v.100.sql - tb\_trn\_rental을 기준으로 tb\_mst\_book,tb\_mst\_user을 left join으로 연결하여 연체 책을 바로 출력할 수 있게 출력하였다.

●insertRentdataAgua.sql - 연체된 책을 출력하기 위해서는 더미 데이터가 필요하기 때문에 더미 책데이터를 넣어주는 sql을 작성하였다. 첫줄은 rollback; 을 넣어서 이전의 잘못된 sql을 다시 되돌 리고 commit;을 매번 넣어서 데이터가 잘 들어가게 하였다.

●insertUserinfoAgura.sql - 사용자의 정보를 임의로 정해서 데이터를 넣는 sql을 만들어서 team1 맴버들과 공유하였다.

●Overdue\_list.java - Overdue\_listServlet에서 필요한 정보를 잘 저장하고 jsp 파일로 잘 보낼수 있게 toString()안에 필요한 정보를 받아오는 순서대로 정렬하여 넣었다. 처음에 dueDate순서가 잘못 되어서 일정 부분이 잘 나오지 못했었으나 다시 정렬을 하여 잘 표현이 될 수 있도록 하였다.



●overdue\_list.jsp - 기존 탬플랫에 있는 리스트 UI 를 지키면서 정보가 잘나오도록 등록을 하고 送信버튼이 催促상태를 기반으로 버튼이 나오던가 나오지 않게 처음에 taglib의 if 함수를 사용하였지만 催促中라도 메일을 보낼 수 있게 버튼을 항상 보이게 하였다. 催促中 의 상태는 true or false 밖에없었지만 이것도 taglib 의 if 함수를 사용하여 true일때는 送信完了, false일 때는 未送信이 나올 수 있게 상태 변경을 하여 사용자가 한번에 催促メール送信状態를 확인 할 수 있게 하였다.

●Overdue\_listServlet.java - DB에서 book\_id,title, author, due\_date, user\_id, name , alert정보를 받아온 후에 list에 넣은 후에 returnList에 넣어서 request.setAttribute("returnList", returnList);를 통해서 returnList를 overview\_list.jsp에 DB에서 받은 정보를 넘겨 주었다. 특히 催促メール送信状態를 바로 적용하기 위해서 status1에 book\_id가 들어오지 않으면 그냥 db에서 상태를 받아오고 book\_id를 받아온다면 executeUpdate를 실행하여 催促メール送信状態를 반영한다.

●Overdue\_send.java - Overdue\_sendServelt.java에서 db의 값을 잘 받아올 수 있게 email, name, title, dueDate, bookId의 setfunction과 getfunction을 만들었다.

●overdue\_send.jsp - book\_id 를 Overdue\_sendServelt.java에서 받아와서 book\_id에 관련된 name, duedate, email을 받아서 자동으로 催促メール을 작성해준다. 거기에 送信을 누르면 다시 Overdue\_listServlet.java에 이동하여 催促メール送信状態반영한 상태에서 期限切れの本のリスト를 출력한다.



●Overdue\_sendServlet.java - book\_id 를 Overdue\_sendServelt.java에서 받아와서 book\_id에 관련된 name, duedate, email을 받아서 overdue\_send.jsp로 보내준다.



●overDueSql\_agura\_ver1.01.sql - 期限切れの本のリスト 표현하기 위해 tb\_trn\_rental을 기준으로 tb\_mst\_book,tb\_mst\_user을 left join으로 연결하여 연체 책을 바로 출력할 수 있게 출력하였다.



●updateAlerttoFalse\_agura.sql - 催促メール送信状態가 DB에 모두 true 상태로 바뀌기 때문에 false 상태로 갱신 시키는 sql



●updateStatus\_agura.sql - 催促メール送信状態를 DB에 설정을 하는 sql