# Hyperledger Fabric Assignment

2017029570 남혜빈

* Item.java

1. Item class는 Item의 이름, 소유주 이름, 상태, 가격 정보를 가지고 있다.

* TradeItem.java

1. registerItem : argument로 등록할 물품의 이름, 소유주 이름을 받아 Item 객체를 생성하고 putStringState를 통해 등록한다. 아직 판매되는 물품은 아니므로 상태 값은 0으로 설정한다.
2. sellMyItem : argument로 판매할 물품의 이름을 받고 이름이 일치하는 물품을 찾아 상태를 1로 바꿔준다. 상태가 1이면 판매중인 물품이라는 의미이다.
3. buyUserItem : argument로 구매할 물품의 이름과 해당 Transcation을 호출한 사용자(구매자) 이름을 받는다. changeItemOwner함수를 호출하여 물품의 소유주를 변경한다.
4. changeItemOwner : argument로 물품의 이름과 새 소유주의 이름을 받는다. 물품의 다른 정보는 유지하고 소유주만 바꿔서 새로운 Item객체를 만들어 등록한다.
5. getMyItem : 해당 Transcation을 호출한 사용자의 이름을 argument로 받아, 등록된 물품 중 소유주가 사용자와 같은 물품을 List에 저장한 후 리턴한다.
6. getAllRegisteredItems : 등록된 물품 정보를 getStateByRange로 구한 후, 반복문을 통해 물품별로 파싱하여 List에 저장한다. 저장한 물품 정보를 리턴한다.
7. getAllOrderedItems : 등록된 물품 정보를 구해 물품의 상태가 등록된 상태가 아닌(status != 0) 물품을 List에 저장한 후 리턴한다.

* query.js

1. 수행할 transaction query가 getMyItems일 경우, evaluateTransaction을 호출할 때 query의 argument로 받은 name을 추가한다. 이는 transaction을 호출한 사용자의 이름이다.
2. transaction이 getMyItems가 아닌 경우는 evaluteTranscation에 query만 넣고 실행한다.
3. 결과를 return 받아 result 변수에 저장한 후, JSON형태로 변환하여 반환한다.

* invoke.js

1. invoke할 함수가 registerItem 또는 buyUserItem일 경우 submitTransaction(함수, 물품 이름, 사용자 이름)으로 invoke한다.
2. 함수가 sellMyItem일 경우는 submitTransaction(함수, 물품 이름)으로 invoke한다.

* routes/index.js

1. main page : main page에서 사용자가 로그인했는지 확인하고 만약 했다면 getMyItems, getAllRegisteredItems, getAllOrderedItems를 요청하고 결과를 rendering한다.
2. /registerItem : 등록할 물품의 이름을 req.body.name에서 추출하여 registerItem함수로 invoke한다.
3. /sellMyItem : 판매할 물품의 이름을 req.body.myItemscategory에서 추출하여 sellMyItem함수로 invoke한다.
4. /buyUserItem : 구매할 물품의 이름을 req.body.salecategory에서 추출하여 buyUserItem함수로 invoke한다.