

프로젝트 1: 요구사항 분석

과목: Database Systems ()

교수: 교수님

조교: 조교님, 조교님

소속:

학번:

이름: 송재현

필수 기능

1. 관리자는 각 사용자의 계좌 입출금 내역을 관리한다.
2. 사용자는 자신의 계좌를 n개까지 개설하거나 삭제한다.
3. 사용자는 자신의 계좌에 돈을 입금할 수 있다.
4. 사용자는 자신의 계좌에서 돈을 출금할 수 있다.

구현이 되면 좋은 기능

1. 계좌간 계좌이체 기능
2. 관리자는 고객의 신용점수를 조절할 수 있다. 관리자 직급에 따른 조절 가능 신용점수 범위 존재한다.
3. 관리자는 직원번호, 이름, 전화번호, 직급 정보를 필수로 기입해야한다. 개설한 계좌, 사수, 부사수 정보는 없어도 된다.
4. 관리자의 사수, 부사수 관계는 recursive relationship이다.
5. 사용자는 고객번호, 이름, 주민번호, 나이, 신용점수, 최대계좌수, 담당관리자 정보가 필수이다. 전화번호, 주소, 직업 정보는 선택이다. 나이는 주민번호를 통한 derived attribute이다. 최대계좌수는 신용점수, 나이에 따라 결정된다.
6. 계좌는 weak entity로 은행상품을 identifying owner로 갖는다. 상품종류, 계좌번호, 사용자번호, 비밀번호, 이자, 책임관리자 정보가 필수이다. 계좌번호는 partial key로 상품종류, 사용자번호, 책임관리자와 함께 key를 이룬다. 입출금내역은 초기에 정보 없음이 가능하고, multi-valued attribute이다.
7. 계좌는 자유입출금 계좌가 필수이고, 다른 계좌(예금, 대출)는 선택 개설할 수 있다.
8. 은행 상품은 상품종류, 상품번호, 이율 정보가 필수이다. 상품마다 신용점수에 따른 이율이 다르다. 따라서 신용점수가 변화하면 추천상품 기능에서 더 나은 예금 혹은 대출 상품을 추천받을 수 있다.

서비스 요구사항 (TEXT 버전)

파란 글씨: relationship

entity: 관리자

attribute: 직원번호(Integer / Key / No NULL), 이름(String / No NULL), 전화번호(Integer / No NULL / Multi-valued), 직급(Integer / No NULL / 1, 2, 3 등급 존재), [개설한계좌](#)(Integer / Yes NULL / 입출금 내역기록_multi-valued), [사수](#)(Integer, Key, Yes NULL, Recursive relationship), [부사수](#)(Integer, Key, Yes NULL, Recursive relationship)

entity: 사용자

attribute: 고객번호(Integer / Key / No NULL), 이름(String / No NULL), 주민번호(Integer / No NULL), 나이(Integer / No NULL / derived attribute), 전화번호(Integer / Yes NULL / multi-valued), 주소(String / Yes NULL / composite attribute), 직업(String / Yes NULL), 신용점수(Integer / No NULL), 최대계좌 수(Integer / No NULL / 신용점수, 나이에 따라 결정), [담당관리자](#)(Integer / Key / No NULL / reference 관리자_직원번호)

entity: 계좌(weak_entity)

attribute: [상품종류](#)(Integer / Key / No NULL / reference 은행상품_상품번호), 계좌번호(Integer / Key / No NULL), [사용자번호](#)(Integer / Key / No NULL), 비밀번호(Integer / No NULL), 이자(Integer / No NULL), [책임관리자](#)(Integer / Key / No NULL / reference 관리자_직원번호), 입출금내역(Integer / Yes NULL / Multi-valued)

entity: 은행상품

attribute: 상품종류(String / No NULL / 자유입출금, 예금, 대출), 상품번호(Integer / Key / No NULL), 이율(Integer / No NULL / 신용점수별 이율이 다름)

서비스 요구사항 (Table 버전)

파란 글씨: relationship

테이블정의서		Database	은행		
테이블명	관리자				
Comment					
Col #	Column Name	Data Type	Key	Null?	Comments
1	직원번호	Integer	Yes	No	
2	이름	String	—	No	
3	전화번호	Integer	—	No	Multi-valued attribute
4	직급	Integer	—	No	1, 2, 3 등급 존재
5	개설한계좌	Integer	—	Yes	입출금내역
6	사수	Integer	Yes	Yes	Recursive relationship
7	부사수	Integer	Yes	Yes	Recursive relationship
테이블명	사용자				
Comment					
Col #	Column Name	Data Type	Key	Null?	Comments
1	고객번호	Integer	Yes	No	
2	이름	String	—	No	
3	주민번호	Integer	—	No	
4	나이	Integer	—	No	Derived attribute
5	전화번호	Integer	—	Yes	Multi-valued attribute
6	주소	String	—	Yes	Composite attribute
7	직업	String	—	Yes	
8	신용점수	Integer	—	No	
9	최대계좌수	Integer	—	No	신용 점수, 나이에 따라 결정
10	담당관리자	Integer	Yes	No	reference 관리자_직원번호
테이블명	계좌				
Comment	Weak entity, 은행상품이 identifying owner				
Col #	Column Name	Data Type	Key	Null?	Comments
1	상품종류	Integer	Yes	No	reference 은행상품_상품번호
2	계좌번호	Integer	Yes	No	Partial Key
3	사용자번호	Integer	Yes	No	reference 사용자_고객번호
4	비밀번호	Integer	—	No	
5	이자	Integer	—	No	
6	책임관리자	Integer	Yes	No	reference 관리자_직원번호
7	입출금내역	Integer	—	Yes	Multi-valued attribute

테이블명	은행상품				
Comment					
Col #	Column Name	Data Type	Key	Null?	Comments
1	상품종류	String	_	No	자유입출금, 예금, 대출
2	상품번호	Integer	Yes	No	
3	이율	Integer	_	No	신용점수별 이율이 다름