

DB NORMALIZATION

```
1. CREATE TABLE user_information (  
    username VARCHAR(255) PRIMARY KEY,  
    upi VARCHAR(255),  
    mobilenumbar VARCHAR(255),  
    email VARCHAR(255)  
);
```

2.

```
CREATE TABLE wallet (  
    wallet_id INT,  
    username VARCHAR2(255),  
    wallet_type VARCHAR2(255),  
    PRIMARY KEY (wallet_id),  
    CONSTRAINT fk_username FOREIGN KEY (username) REFERENCES user_information(username)  
);
```

```
CREATE SEQUENCE wallet_id_seq START WITH 1 INCREMENT BY 1;
```

```
CREATE OR REPLACE TRIGGER wallet_id_trigger  
BEFORE INSERT ON wallet  
FOR EACH ROW  
BEGIN  
    SELECT wallet_id_seq.NEXTVAL INTO :new.wallet_id FROM dual;  
END;  
/
```

```
3. CREATE TABLE recharge_information (  
    recharge_id INT,  
    username VARCHAR2(255),
```

```
recharged_date DATE,  
recharged_provider VARCHAR2(255),  
recharged_to VARCHAR2(255),  
recharged_amount DECIMAL(10, 2),  
PRIMARY KEY (recharge_id),  
CONSTRAINT fk_recharge_username FOREIGN KEY (username) REFERENCES  
user_information(username)  
);
```

```
CREATE SEQUENCE recharge_id_seq START WITH 1 INCREMENT BY 1;
```

```
CREATE OR REPLACE TRIGGER recharge_id_trigger  
BEFORE INSERT ON recharge_information  
FOR EACH ROW  
BEGIN  
    SELECT recharge_id_seq.NEXTVAL INTO :new.recharge_id FROM dual;  
END;  
/
```

SYSTEM.USER_INFORMATION		
P *	USERNAME	VARCHAR2 (255 BYTE)
	UPI	VARCHAR2 (255 BYTE)
	MOBILENUMBER	VARCHAR2 (255 BYTE)
	EMAIL	VARCHAR2 (255 BYTE)
USER_INFORMATION_PK (USERNAME)		

SYSTEM.RECHARGE_INFORMATION		
P *	RECHARGE_ID	NUMBER (*,0)
F	USERNAME	VARCHAR2 (255 BYTE)
	RECHARGED_DATE	DATE
	RECHARGED_PROVIDER	VARCHAR2 (255 BYTE)
	RECHARGED_TO	VARCHAR2 (255 BYTE)
	RECHARGED_AMOUNT	NUMBER (10,2)
RECHARGE_INFORMATION_PK (RECHARGE_ID)		
FK_RECHARGE_USERNAME (USERNAME)		

SYSTEM.WALLET		
P *	WALLET_ID	NUMBER (*,0)
F	USERNAME	VARCHAR2 (255 BYTE)
	WALLET_TYPE	VARCHAR2 (255 BYTE)
WALLET_PK (WALLET_ID)		
FK_USERNAME (USERNAME)		

