**Practical – 3**

**Aim: Write a program to queue multiple transactions created by clients in blockchain application**

**Code:**

***main.rs***

use simple\_blockchain\_rs::client::Client;

use simple\_blockchain\_rs::transaction::Transaction;

fn main() {

let mut transactions = vec];

let utsav = Client::new();

let bhupendra = Client::new();

let jash = Client::new();

println!("utsav public key: {}", utsav.identify());

println!("bhupendra public key: {}", bhupendra.identify());

println!("jash public key: {}", jash.identify());

println!("");

let mut transaction1 = Transaction::new(utsav.public\_key, bhupendra.public\_key, 10.0, None);

transaction1.sign\_transaction(&utsav);

println!(

"Transaction 1 signature validation: {:#?}",

transaction1.is\_valid\_transaction()

);

transactions.push(transaction1);

let mut transaction2 = Transaction::new(bhupendra.public\_key, jash.public\_key, 10.0, None);

transaction2.sign\_transaction(&bhupendra);

println!(

"Transaction 2 signature validation: {:#?}",

transaction2.is\_valid\_transaction()

);

transactions.push(transaction2);

let mut transaction3 = Transaction::new(jash.public\_key, utsav.public\_key, 10.0, None);

transaction3.sign\_transaction(&jash);

println!(

"Transaction 3 signature validation: {:#?}",

transaction3.is\_valid\_transaction()

);

transactions.push(transaction3);

println!("");

transactions.iter().for\_each(|transaction| {

transaction.print\_transaction();

});

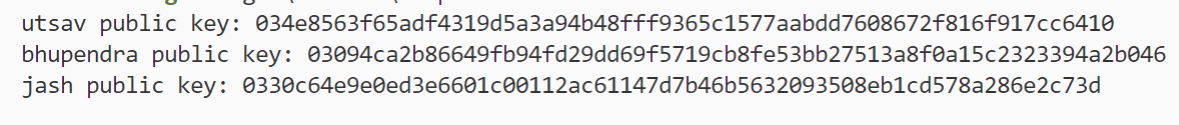
***client.rs from practical - 1***

***transaction.rs from practical – 2***

***lib.rs from practical – 2***

***Transactions between Utsav, Bhupendra and Jash as three clients***

**Clients public keys**



**Transaction between Utsav – Bhupendra**

![Graphical user interface, text

Description automatically generated

**Transaction between Bhupendra – Jash**

Text

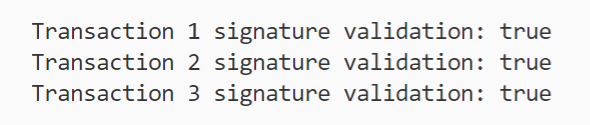
Description automatically generated with medium confidence

**Transaction between Jash – Utsav**

Text

Description automatically generated

**Signature Verification**



**Complete output on stdout**

Graphical user interface, text, application

Description automatically generated