Practical No. 4

```
import java.util.*;
import java.net.*;
import java.io.*;
class BerkeleyClockSynchronization {
static final int PORT = 12345;
static final int N = 5;  // Number of clients
static long coordinatorTime = 0;  // The time of the coordinator
static List<ClientHandler> clients = new ArrayList<>();
static int totalDelay = 0;
public static void main(String[] args) throws IOException {
ServerSocket serverSocket = new ServerSocket(PORT);
System.out.println("Coordinator started, waiting for clients...");
// Accepting N clients
for (int i = 0; i < N; i++) {
Socket clientSocket = serverSocket.accept();
ClientHandler client = new ClientHandler(clientSocket, i);
clients.add(client);
new Thread(client).start();
}
// Coordinator time is set (simulate the coordinator's clock)
coordinatorTime = System.currentTimeMillis();
// Wait for all clients to report back
try {
```

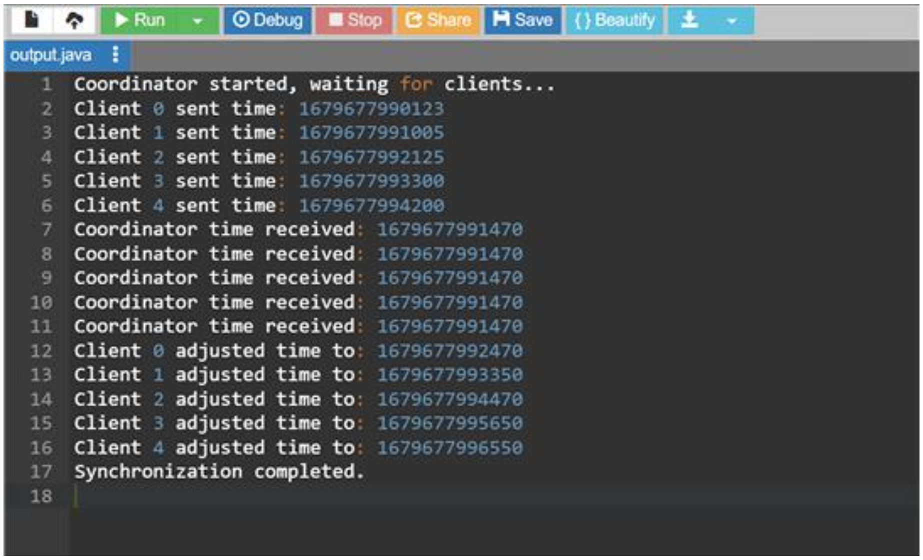
```
Thread.sleep(2000);
} catch (InterruptedException e) {
e.printStackTrace();
}
// Compute the time adjustments for each client
int offsetSum = 0;
for (ClientHandler client : clients) {
offsetSum += client.getOffset();
}
long avgOffset = offsetSum / N;
// Send back the corrected time to clients
for (ClientHandler client : clients) {
client.sendCorrectedTime(avgOffset);
}
System.out.println("Synchronization completed.");
serverSocket.close();
}
// ClientHandler: handles each client's request
static class ClientHandler implements Runnable {
private Socket clientSocket;
private int clientId;
private long clientTime;
private int offset;
```

```
public ClientHandler(Socket socket, int id) {
this.clientSocket = socket;
this.clientId = id;
}
```

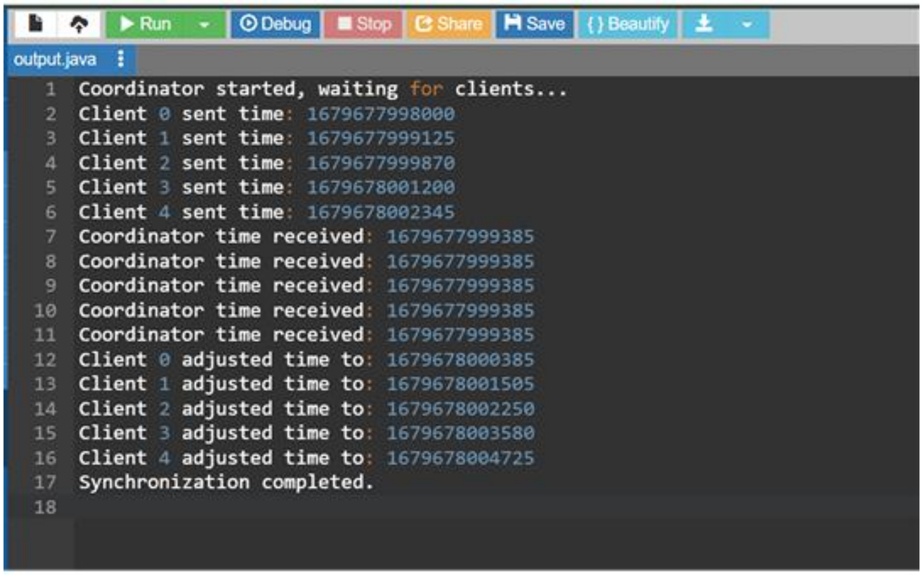
```
@Override
public void run() {
try {
DataInputStream input = new
DataInputStream(clientSocket.getInputStream());
DataOutputStream output = new
DataOutputStream(clientSocket.getOutputStream())
;
// Client sends its current time to the coordinator
clientTime = System.currentTimeMillis();
output.writeLong(clientTime);
System.out.println("Client " + clientId + " sent time:
" + clientTime);
// Coordinator sends its time
long coordinatorReceivedTime = input.readLong();
System.out.println("Coordinator time received: " +
coordinatorReceivedTime);
// Calculate the offset between client and
coordinator
offset = (int) (clientTime -
coordinatorReceivedTime);
// Send the offset back to the coordinator
output.writeInt(offset);
// Wait for corrected time from the coordinator
long correctedTime = input.readLong();
```

```
long finalClientTime = clientTime + correctedTime;
System.out.println("Client " + clientId + " adjusted
time to: " + finalClientTime);
input.close();
output.close();
clientSocket.close();
} catch (IOException e) {
e.printStackTrace();
}
}
public int getOffset() {
return offset;
}
public void sendCorrectedTime(long avgOffset)
throws IOException {
DataOutputStream output = new
DataOutputStream(clientSocket.getOutputStream())
;
output.writeLong(avgOffset);
output.writeLong(coordinatorTime);
System.out.println("Sent corrected time " +
coordinatorTime + " to client " +
clientId);
}
}
}
```

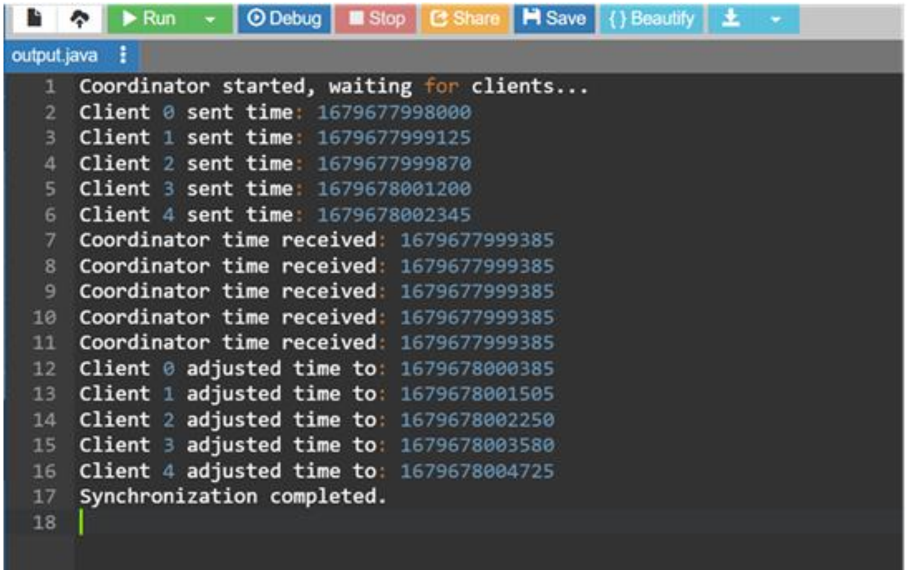
Output :



```
Coordinator started, waiting for clients...
Client 0 sent time: 1679677990123
Client 1 sent time: 1679677991005
Client 2 sent time: 1679677992125
Client 3 sent time: 1679677993300
Client 4 sent time: 1679677994200
Coordinator time received: 1679677991470
Coordinator time received: 1679677991470
Coordinator time received: 1679677991470
Coordinator time received: 1679677991470
Coordinator time received: 1679677991470
Client 0 adjusted time to: 1679677992470
Client 1 adjusted time to: 1679677993350
Client 2 adjusted time to: 1679677994470
Client 3 adjusted time to: 1679677995650
Client 4 adjusted time to: 1679677996550
Synchronization completed.
```



```
Coordinator started, waiting for clients...
Client 0 sent time: 1679677998000
Client 1 sent time: 1679677999125
Client 2 sent time: 1679677999870
Client 3 sent time: 1679678001200
Client 4 sent time: 1679678002345
Coordinator time received: 1679677999385
Coordinator time received: 1679677999385
Coordinator time received: 1679677999385
Coordinator time received: 1679677999385
Coordinator time received: 1679677999385
Client 0 adjusted time to: 1679678000385
Client 1 adjusted time to: 1679678001505
Client 2 adjusted time to: 1679678002250
Client 3 adjusted time to: 1679678003580
Client 4 adjusted time to: 1679678004725
Synchronization completed.
```



```
Coordinator started, waiting for clients...
Client 0 sent time: 1679677998000
Client 1 sent time: 1679677999125
Client 2 sent time: 1679677999870
Client 3 sent time: 1679678001200
Client 4 sent time: 1679678002345
Coordinator time received: 1679677999385
Coordinator time received: 1679677999385
Coordinator time received: 1679677999385
Coordinator time received: 1679677999385
Coordinator time received: 1679677999385
Client 0 adjusted time to: 1679678000385
Client 1 adjusted time to: 1679678001505
Client 2 adjusted time to: 1679678002250
Client 3 adjusted time to: 1679678003580
Client 4 adjusted time to: 1679678004725
Synchronization completed.
```