

## Java-Beispiel Ticketbestellung (2)

### Java-Beispiel Ticketbestellung (1)

1. Client „bestellt“ beim Server ein Ticket
2. Server liefert Client eine Ticketnummer zurück

```
10 import java.rmi.*;
20 public interface TicketBestellung extends Remote {
30     public int getNextTicket(String name) throws RemoteException;
40 }
```

```
100 import java.rmi.*;
105 import java.rmi.server.UnicastRemoteObject;
110 public class TicketServer extends UnicastRemoteObject implements TicketBestellung {
120     int nextTicket = 0;
130     public TicketServer() throws RemoteException {};
140     public int getNextTicket(String name) throws RemoteException {
150         return nextTicket++;
160     }
170     public static void main(String [] args) {
180         try {
190             String name = "//pepita.cs.tu-berlin.de/" + "Benutzername";
200             TicketServer server = new TicketServer();
210             Naming.bind(name, server);
220         }
230         catch(Exception e) {System.out.println("FEHLER!");}
240     }
250 }
```

### Java-Beispiel Ticketbestellung (3)

```
300 import java.rmi.*;
310 public class TicketClient {
320     public static void main(String [] args) {
330         String serverHost = "pepita.cs.tu-berlin.de";
340         String serverName = args[0];
350         String fullname = "rmi://" + serverHost + "/" + serverName;
360         TicketBestellung server = null;
370         try {
380             server = (TicketBestellung) Naming.lookup(fullname);
390         }
400         catch (Exception e) {System.out.println("FEHLER bei Lookup!");}
410
420         try {
430             int ticket = server.getNextTicket("TicketClient");
440             System.out.println("Ticket " + ticket + " erhalten!");
450         }
460         catch (Exception e) {System.out.println("FEHLER!");}
470     }
```