

```

1 package Tests;
2
3 import static org.junit.Assert.*;
18
19 public class ReservierungsTest {
20
21     private IPersistenzService pServ;
22     private IGastServices gastServ;
23     private IReservierungServices resvServ;
24     private IGastServicesFuerReservierung gastServFuerResrv;
25     private GastTyp swaneet, steffen;
26     private ZusatzleistungTyp wlan, balkon;
27     private int zNr;
28
29     @Before
30     public void setUp() {
31         try {
32             this.pServ = new DatabaseConnection();
33             this.gastServ = new GastverwaltungKomponente(pServ);
34             this.gastServFuerResrv = new GastverwaltungKomponenteDummy(pServ);
35             this.resvServ = new Reservierungskomponente(pServ,
36                 this.gastServFuerResrv);
37
38             // create guests and zusatzleistungen for testing
39             this.swaneet = gastServ.erzeugeGast(1, "Swaneet",
40                 email("swaneet", "hawhamburg", "de"));
41             this.steffen = gastServ.erzeugeGast(2, "Steffen",

```

```

42         email("steffer", "hawhamburg", "de"));
43
44         this.wlan = resvServ.erzeugeZusatzleistung("WLAN");
45         this.balkon = resvServ.erzeugeZusatzleistung("balkon");
46         zNr = 0;
47     } catch (Exception e) {
48         e.printStackTrace();
49     }
50 }
51
52 @Test
53 public void testReservierungIsolated() {
54     try {
55         swaneet = gastServ.sucheGastNachName("Swaneet");
56         steffen = gastServ.sucheGastNachName("Steffen");
57
58         assertFalse(swaneet.istStammkunde());
59         assertFalse(steffen.istStammkunde());
60
61         // swaneet macht drei reservierungen mit zusatzleistung
62         for (int i = 1; i <= 3; i++) {
63             ReservierungTyp res = resvServ.reserviereZimmer(swaneet.nr(),
64                 zimmerNr());
65             resvServ.bucheZusatzleistung(res.nr(), balkon.nr());
66             resvServ.bucheZusatzleistung(res.nr(), wlan.nr());
67         }
68

```

```

69         ReservierungTyp res = resvServ.reserviereZimmer(steffen.nr(),
70             zimmerNr());
71         resvServ.bucheZusatzleistung(res.nr(), wlan.nr());
72
73         swaneet = gastServ.sucheGastNachName("Swaneet");
74         steffen = gastServ.sucheGastNachName("Steffen");
75
76         assertTrue(swaneet.istStammkunde());
77         assertFalse(steffen.istStammkunde());
78     } catch (Exception e) {
79         e.printStackTrace();
80     }
81 }
82
83 // gibt immer eine neue zimmerNr zur.ck
84 private int zimmerNr() {
85     zNr += 1;
86     return zNr;
87 }
88 }
89

```