

9. Aufgabenblatt

CORBA

JacORB

- <http://www.jacorb.org/>
- `/import/public/opt/JacORB222/bin`
- `cp /import/public/opt/JacORB222/etc/orb.properties ~`
nicht zwingend aber u.a. Konfiguration des Namensdienst

```

module phonebook {
    // Datei phonebook/Book.idl
    // -> Java-Package phonebook

    interface Book;
    // Vorwärtsdeklaration (hier obsolet)

    struct Entry {
        // zusammengesetzter Datentyp
        // (nicht fernaufrufbar)
        string name;
        string number;
    };

    typedef sequence<Entry> Entries; // sequence nur in typedef oder structs
    // verwendbar, nicht z.B. als Rückgabewert

    exception NoSuchEntry {
        // -> Java-Exception
        string name;
    };

    interface Book {
        void enter(in Entry e);
        string lookup(in string name) raises (NoSuchEntry);
        Entries list();
    };
};

```

IDL-Compiler

- `[/import/public/opt/JacORB222/bin/] idl phonebook/Book.idl`

Book.java

BookHelper.java

BookHolder.java

BookOperations.java

BookPOA.java

BookPOATie.java

EntriesHelper.java

EntriesHolder.java

Entry.java

EntryHelper.java

EntryHolder.java

NoSuchEntry.java

NoSuchEntryHelper.java

NoSuchEntryHolder.java

_BookStub.java

- **Zu implementieren:**

BookImpl.java

BookServer.java

Client.java

```
package phonebook;
```

```
import java.util.HashMap;
```

```
public class BookImpl extends BookPOA {
```

```
    private final HashMap book = new HashMap();
```

```
    public void enter(Entry e) {  
        book.put(e.name, e);  
    }
```

```
    public String lookup(String name) throws NoSuchEntry {  
        [...]  
    }
```

```
    public Entry[] list() {  
        [...]  
    }  
}
```

```
import java.io.*; import org.omg.CORBA.*; import org.omg.PortableServer.*;

public class BookServer { // Usage: jaco phonebook.BookServer <ior_file>
    public static void main(String[] args) throws Exception {

        //init ORB & POA
        ORB orb = ORB.init( args, null );
        POA poa = POAHelper.narrow( orb.resolve_initial_references( "RootPOA" ));
        poa.the_POAManager().activate();

        // create a BookImpl object
        BookImpl bookImpl = new BookImpl();

        // create the object reference
        org.omg.CORBA.Object obj = poa.servant_to_reference(bookImpl);

        // print stringified object reference to file (alt.: use name service)
        PrintWriter pw = new PrintWriter(new FileWriter(args[0]));
        pw.println( orb.object_to_string( obj ));
        pw.flush(); pw.close();

        // wait for requests
        orb.run();
    }
}
```

```
package phonebook;
import java.io.*; import org.omg.CORBA.*;

public class Client { // Usage: jaco phonebook.Client <ior_file>
    public static void main(String args[]) throws Exception {
        // initialize the ORB.
        ORB orb = ORB.init(args, null);

        // get object reference from command-line argument file
        BufferedReader br = new BufferedReader(new FileReader(args[0]));
        org.omg.CORBA.Object obj = orb.string_to_object(br.readLine());
        br.close();

        // and narrow it to Book
        // if this fails, a BAD_PARAM will be thrown
        Book book = BookHelper.narrow(obj);

        // use the book object
        book.enter(new Entry("max", "12345"));
        book.enter(new Entry("karsten", "23456"));
        System.out.println("lookup max: " + book.lookup("max"));
    }
}
```