

Information System Design (U08182)

Tutorial Exercise 4 Answer

1. List the different categories of software that may be used in developing a system.
CASE tools; compilers, interpreters and run-time support; visual editors; integrated development environments; class browsers; component managers; DBMS; CORBA; testing tools; conversion tools; documentation generators; configuration management systems; debuggers.

2. What packages (not the UML package concept) have you used and what categories do they fall into?

The answer to this will depend on the individual. Examples would be:

Rational Rose – CASE tool;

Java – compiler with run-time support;

JBuilder – visual editor/IDE;

Microsoft Developer's Studio – IDE;

Oracle – DBMS;

VisiBroker – CORBA ORB;

TestScope – testing tool;

ClearCase – configuration management tool;

gdb – debugger.

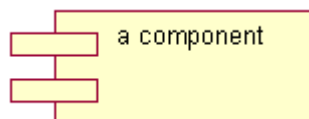
3. Define what is meant by a component.

A physical element of a system, usually a file of some sort. can be a source file, used in producing the software for the system, or an element of the runtime system

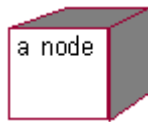
4. Define what is meant by a node.

Processor in the implementation of a system, shown in a deployment diagram.

5. What is a notation for a component?



6. What is a notation for a node?



7. What are the main purposes of using component diagrams?

Model physical software components and the relationships between them;

Model source code and relationships between files;

Model the structure of releases of software;

Specify the files that are compiled into an executable.

8. What are the main purposes of using deployment diagrams?

Model physical hardware elements and the communication paths between them;

Plan the architecture of the system;

Document the deployment of software components on hardware nodes.

9. What is the difference between components in a component diagram and components in a deployment diagram?

Components shown in a deployment diagram model the deployment of run-time components on processors in the planned system. They are normally instances of components. Components in a component diagram are usually source files or other elements of the system software under development.

10. How are dependencies shown in a component diagram?

As a dashed arrow with an open arrow head.

11. What is the difference between a package diagram and a component diagram?

Package diagrams show the logical grouping of classes in a system, whereas component diagrams show the physical components of a system. During implementation, package diagrams can be used to show the grouping of physical components into sub systems; component diagrams can be combined with deployment diagrams to show the physical location of components of the system.

12. What are the main steps in producing a component diagram?

1. Decide on the purpose of the diagram;

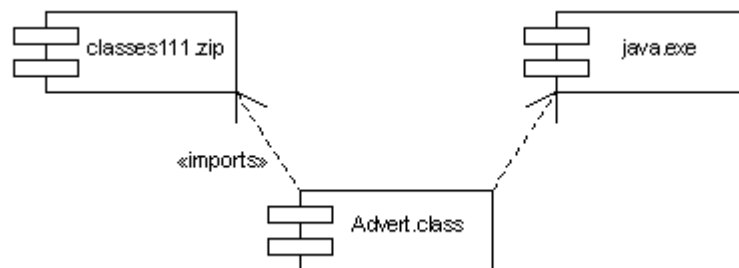
2. Add components to the diagram grouping them within other components if appropriate;

3. Add other elements to the diagram, such as classes objects or interfaces;

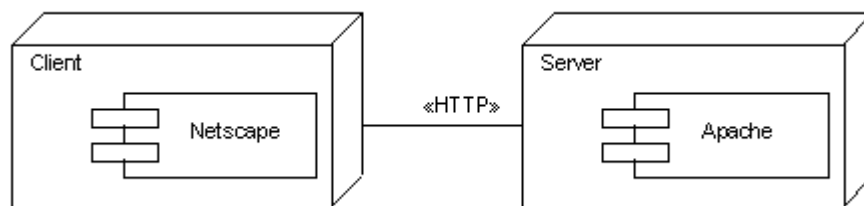
4. Add the dependencies between the elements of the diagram.
13. What are the main steps in producing a deployment diagram?
 1. Decide on the purpose of the diagram;
 2. Add nodes to the diagram;
 3. Add communication associations to the diagram;
 4. Add other elements to the diagram, such as components or active objects, if required;
 5. Add dependencies between components and objects, if required.
 14. What are the benefits of reuse?

Saving time and money in developing the components, and saving time and money in testing the components.
 15. What are some of the obstacles of reuse?

Inappropriate choice of projects for reuse; planning for reuse too late in a project; the level of coupling between classes in an object-oriented design; the lack of standards for reusable components.
 16. Draw a component diagram to show the run-time dependency between a Java class file, the java.exe run-time program and the Java classes in a Zip file.



17. Draw a deployment diagram to show how a Web browser and a Web server are located on different machines and the communication protocol they use.



18. A database contains the following tables.

Hotel (hotelNo, hotelName, city, address)

Room (row, roomNo, hotelNo, roomType, price)

- A) Write SQL statements to produce, firstly a table of all towns/cities that have hotels; secondly a table of hotels that are in a specified town/city, of a particular room type, and within a price range.

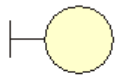
```
SELECT Hotel.city  
FROM Hotel;
```

```
SELECT Room.type,  
       Room.price,  
       Hotel.hotelName,  
       Hotel.city,  
       Hotel.address  
FROM Hotel INNER JOIN Room ON Hotel.hotelNo=Room.hotelNo  
WHERE Hotel.city='London'  
And Room.type='single'  
And Room.price<=100;
```

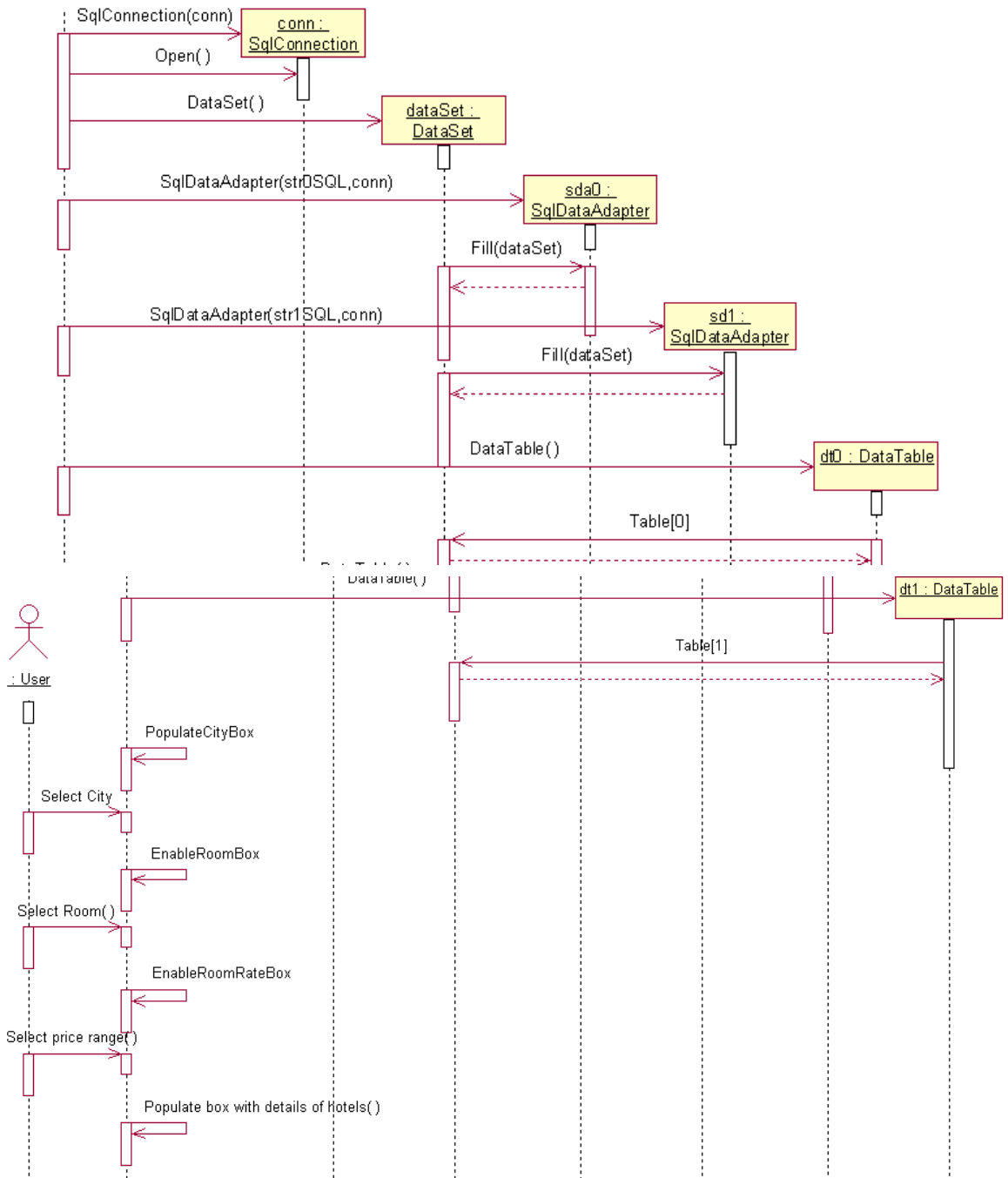
- B) Figure 1 illustrates an interface used to locate a hotel based upon the criteria given in the previous part of the question. Assuming ADO.net is being used, draw a sequence diagram to show how the interface is populated with information from the database tables, to enable a user to make an informed choice of hotel.



Figure 1: GUI Hotel Finder

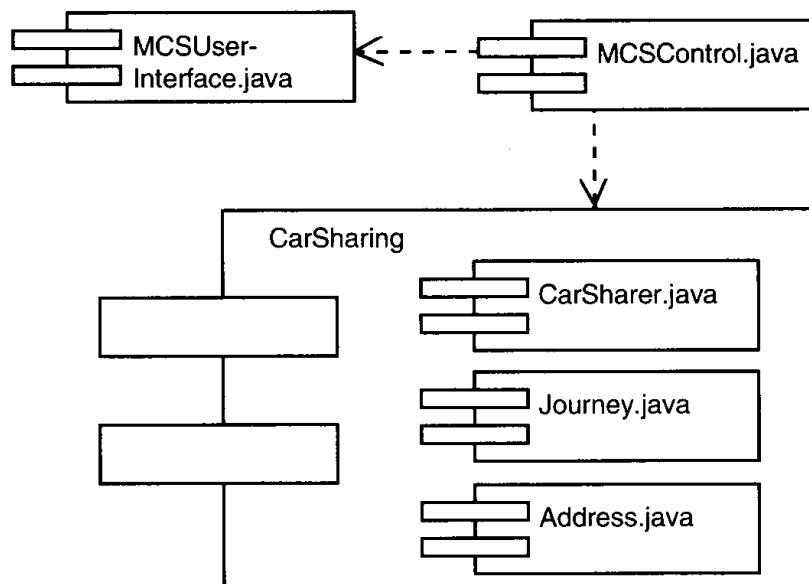


: HotelFinder

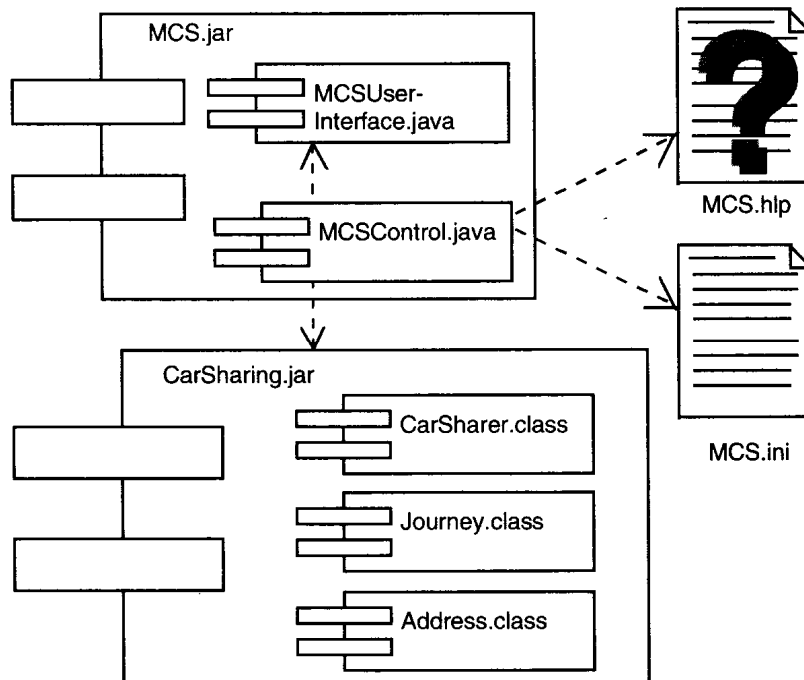


19. Three entity classes are used in a collaboration – CarSharer, Journey and Address. Each of these classes will be implemented by a (.java) source file. These classes are used across a number of use cases and are grouped together into a CarSharing component as Java (.class) files. here we are just dealing with the source files. There are two other classes MCSUserInterface and MCSCControl. Each of these will be implemented by a (.java) file. The MCSCControl component has a dependency on the CarSharing component and on the MCSUserInterface component

A) Draw a component diagram showing the source code dependencies. The .class files are grouped together into two Java archive (.jar) files. The MCSCControl.class component will need to read a configuration file (MCS.ini) and display a help file (MCS.hlp) when required. The MCSCControl (.class) file also has dependencies on the MCSUserInterface (.java) file and the CarSharing (.jar) components.



B) Re-draw the component diagram to show the run-time component dependencies.



20. Draw a deployment diagram given that the nodes are three client PCs, a server and a printer. The communications protocol between the clients and server is TCP/IP; and between the server and the printer is a standard parallel printer protocol. The user interface and the control objects will run on the clients.

