Chapter 8 Questions

**1.** What are the basic characteristics of object-oriented

systems?

**2.** What is dynamic binding?

**3.** Define polymorphism. Give one example of a good

use of polymorphism and one example of a bad use of

polymorphism.

**4.** What is an inheritance conflict? How does an inheritance

conflict affect the design?

**5.** Why is cancellation of methods a bad thing?

**6.** Give the guidelines to avoid problems with inheritance

conflicts.

**7.** How important is it to know which object-oriented

programming language is going to be used to implement

the system?

**8.** What additional types of inheritance conflicts are

there when using multiple inheritance?

**9.** What is the law of Demeter?

**10.** What are the six types of interaction coupling? Give

one example of good interaction coupling and one

example of bad interaction coupling.

**11.** What are the seven types of method cohesion? Give

one example of good method cohesion and one

example of bad method cohesion.

**12.** What are the four types of class cohesion? Give one

example of each type.

**13.** What are the five types of connascence described in

your text? Give one example of each type.

**14.** When designing a specific class, what types of additional

specification for a class could be necessary?

**15.** What are exceptions?

**16.** What are constraints? What are the three different

types of constraints?

**17.** What are patterns, frameworks, class libraries, and

components? How are they used to enhance the

evolving design of the system?

**18.** How are factoring and normalization used in designing

an object system?

**19.** What are the different ways to optimize an object

system?

**20.** What is the typical downside of system optimization?

**21.** What is the purpose of a contract? How are contracts

used?

**22.** What is the Object Constraint Language? What is its

purpose?

**23.** What is the Structured English? What is its purpose?

**24.** What is an invariant? How are invariants modeled in a

design of a class? Give an example of an invariant for

a hourly employee class using the Object Constraint

Language.

**25.** Create a contract for a compute pay method associated

with an hourly employee class. Specify the preconditions

and postconditions using the Object

Constraint Language.

**26.** How do you specify a method’s algorithm? Give an

example of an algorithm specification for a compute

pay method associated with an hourly employee class

using Structured English.

**27.** How do you specify a method’s algorithm? Give an

example of an algorithm specification for a compute

pay method associated with an hourly employee class

using an activity diagram.

**28.** How are methods specified? Give an example of a

method specification for a compute pay method associated

with an hourly employee class.

Chapter 9

**1.** Describe the four steps in object persistence design.

**2.** How are a file and a database different from each other?

**3.** What is the difference between an end-user database

and an enterprise database? Provide an example of

each one.

**4.** What are the differences between sequential and random

access files?

**5.** Name five types of files and describe the primary purpose

of each type.

**6.** What is the most popular kind of database today?

Provide three examples of products that are based on

this database technology.

**7.** What is referential integrity and how is it implemented

in an RDBMS?

**8.** List some of the differences between an ORDBMS and

an RDBMS.

**9.** What are the advantages of using an ORDBMS over

an RDBMS?

**10.** List some of the differences between an ORDBMS and

an OODBMS.

**11.** What are the advantages of using an ORDBMS over

an OODBMS?

**12.** What are the advantages of using an OODBMS over

an RDBMS?

**13.** What are the advantages of using an OODBMS over

an ORDBMS?

**14.** What are the factors in determining the type of object

persistence format that should be adopted for a

system? Why are these factors so important?

**15.** Why should you consider the storage formats that

already exist in an organization when deciding upon a

storage format for a new system?

**16.** When implementing the object persistence in an

ORDBMS, what types of issues must you address?

**17.** When implementing the object persistence in an

RDBMS, what types of issues must you address?

**18.** Name three ways null values can be interpreted in a

relational database. Why is this problematic?

**19.** What are the two dimensions in which to optimize a

relational database?

**20.** What is the purpose of normalization?

**21.** How does a model meet the requirements of third

normal form?

**22.** Describe three situations that can be good candidates

for denormalization.

**23.** Describe several techniques that can improve performance

of a database.

**24.** What is the difference between interfile and intrafile

clustering? Why are they used?

**25.** What is an index and how can it improve the performance

of a system?

**26.** Describe what should be considered when estimating

the size of a database.

**27.** Why is it important to understand the initial and

projected size of a database during design?

**28.** What are some of the nonfunctional requirements that

can influence the design of the data management layer?

**29.** What are the key issues in deciding between using

perfectly normalized databases and denormalized

databases?

**30.** What is the primary purpose of the data access and

manipulation classes?

**31.** Why should the data access and manipulation classes

be dependent on the problem domain classes instead

of the other way around?

**32.** Why should the object persistence classes be dependent

on the problem domain classes instead of the

other way around?

Chapter 10

**1.** Explain three important user interface design

principles.

**2.** What are three fundamental parts of most user interfaces?

**3.** Why is content awareness important?

**4.** What is white space, and why is it important?

**5.** Under what circumstances should densities be low?

High?

**6.** How can a system be designed to be used by both

experienced and first-time users?

**7.** Why is consistency in design important? Why can too

much consistency cause problems?

**8.** How can different parts of the interface be consistent?

**9.** Describe the basic process of user interface design.

**10.** What are use cases, and why are they important?

**11.** What is a WND, and why is it used?

**12.** Why are interface standards important?

**13.** Explain the purpose and contents of interface metaphors,

interface objects, interface actions, interface

icons, and interface templates.

**14.** Why do we prototype the user interface design?

**15.** Compare and contrast the three types of interface

design prototypes.

**16.** Why is it important to perform an interface evaluation

before the system is built?

**17.** Compare and contrast the four types of interface

evaluation.

**18.** Under what conditions is heuristic evaluation justified?

**19.** What type of interface evaluation did you perform in

the Your Turn 10-1?

**20.** What are Krug’s three design principles?

**21.** Describe three basic principles of navigation design.

**22.** How can you prevent mistakes?

**23.** Explain the differences between object-action order

and action-object order.

**24.** Describe four types of navigation controls

**25.** Why are menus the most commonly used navigation

control?

**26.** Compare and contrast four types of menus.

**27.** Under what circumstances would you use a dropdown

menu versus a tab menu?

**28.** Under what circumstances would you use an image

map versus a simple list menu?

**29.** Describe five types of messages.

**30.** What are the key factors in designing an error message?

**31.** What is context-sensitive help? Does your word

processor have context-sensitive help?

**32.** How do an essential use case and a real use case differ?

**33.** What is the relationship between essential use cases

and use scenarios?

**34.** What is the relationship between real use cases and use

scenarios?

**35.** Explain three principles in the design of inputs.

**36.** Compare and contrast batch processing and online processing.

Describe one application that would use batch

processing and one that would use online processing.

**37.** Why is capturing data at the source important?

**38.** Describe four devices that can be used for source data

automation.

**39.** Describe five types of inputs.

**40.** Compare and contrast check boxes and radio buttons.

When would you use one versus the other?

**41.** Compare and contrast on-screen list boxes and dropdown

list boxes. When would you use one versus the

other?

**42.** Why is input validation important?

**43.** Describe five types of input validation methods.

**44.** Explain three principles in the design of outputs.

**45.** Describe five types of outputs.

**46.** When would you use electronic reports rather than

paper reports, and vice versa?

**47.** What do you think are three common mistakes that

novice analysts make in navigation design?

**48.** What do you think are three common mistakes that

novice analysts make in input design?

**49.** What do you think are three common mistakes that

novice analysts make in output design?

**50.** How would you improve the form in Figure 10-4?

**51.** What are the six challenges you face when developing

mobile applications?

**52.** What are the six suggestions to address the mobile

computing challenges?

**53.** With regard to social media, what is the difference

between “push” and “pull” approaches to interacting

with customers?

**54.** Why is it important to keep your social media sites

synced?

**55.** How can you keep your customers engaged with your

social media sites?

**56.** What are some of the multilingual issues that you may

face when developing for a global audience?

**57.** How important is the proper use of color when developing

websites for a global audience? Give some examples

of potential pitfalls that you could run into.

**58.** Name the three cultural dimensions that are relevant

to user interface design identified by Hall. Why are

they relevant?

**59.** Name the four cultural dimensions that are relevant to

user interface design identified by Hofstede. Why are

they relevant?

**60.** What are some of the nonfunctional requirements

that can influence the design of the human–computer

interaction layer?

Chapter 11

**1.** What are the four basic functions of any information

system?

**2.** What are the three primary hardware components of

any physical architecture?

**3.** Name two examples of a server.

**4.** Compare and contrast server-based architectures,

client-based architectures, and client–server-based

architectures.

**5.** What is the biggest problem with server-based computing?

**6.** What is the biggest problem with client-based computing?

**7.** Describe the major benefits and limitations of thin

client–server architectures.

**8.** Describe the major benefits and limitations of thick

client–server architectures.

**9.** Describe the differences among two-tiered, threetiered,

and *n*-tiered architectures.

**10.** Define *scalable.* Why is this term important to system

developers?

**11.** What six criteria are helpful to use when comparing

the appropriateness of computing alternatives?

**12.** Why should the project team consider the existing

physical architecture in the organization when designing

the physical architecture layer of the new system?

**13.** Name the three different types of clouds. How do they

differ from one another?

**14.** What is meant by a service-oriented architecture?

**15.** Define virtualization. How does it relate to the cloud?

**16.** What are the differences among IaaS, PaaS, and SaaS?

**17.** What are the obstacles for provisioning the physical

architecture layer with cloud technologies?

**18.** What, if any, are the issues related to security in the

cloud?

**19.** What are SOX and HIPAA and how could they affect

a firm’s decision to adopt cloud technology?

**20.** How do tablets, such as the iPadTM, enable the paperless

office?

**21.** What additional hardware- and software-associated

costs might need to be included on the hardware and

software specification?

**22.** Who is ultimately in charge of acquiring hardware

and software for a project?

**23.** What is a benchmark and why is it important?

**24.** Why is Parkinson’s Law relevant to the design of the

physical architecture layer?

**25.** What do you think are three common mistakes that

novice analysts make in architecture design and hardware

and software specification?

**26.** Describe the major nonfunctional requirements and

how they influence physical architecture layer

design.

**27.** Why is it useful to define the nonfunctional requirements

in more detail even if the technical environment

requirements dictate a specific architecture?

**28.** What does the network model communicate to the

project team?

**29.** What are the differences between the top-level network

model and the low-level network model?

**30.** Are some nonfunctional requirements more important

than others in influencing the architecture design

and hardware and software specification?

**31.** What do you think are the most important security

issues for a system?