**41 review questions**

**(from question 3-43)**

**3. What is a “thing” called in models used by tradi-**

**tional analysts and database analysts?**

* Data entity.

**4. What is a “thing” called in newer approaches that**

**use UNL?**

* Object or object class.

**5. What are two techniques for identifying things in**

**the problem domain?**

* Brainstorming: a technique to identify problem domain objects in which developers work with users in an open group setting
* Noun technique: a technique to identify problem domain objects by finding and classifying the nouns in a dialog or description.

**6. What are some examples of tangible things in the**

**problem domain of a restaurant?**

* Dishes, cutlery, cookware, prepware, stoves, ovens, frozen foods, refrigerator, dried foods, cleanning supplies

**7. What are some sites or locations in the problem**

**domain of a restaurant?**

* Dinning area, kitchen, service desk, office.

**8. What are some roles played by people in the problem domain of a restaurant?**

* Chef, waiter, waitress, bartender, customer

**9. What are the main steps of the brainstorming**

**technique?**

* Identify a user and a set of use cases.
* Brainstorm with the user to identify things involved when carrying out the use case - that is, things, about which information should be captured by the system.
* Use the types of things to systematically ask questions about potential things, such as the following: are there any tangible things you store information about?
* Continue to work with all types of users and stakeholders to expand the brainstorming list.
* Merge the results, eliminate any duplicates, and compile an initial list.

**10. Explain why identifying nouns helps identify**

**things in the problem domain.**

* A noun is a person, place or thing, therefore identifying nouns might help you identify what needs to be stored by the system.

**11. What are the main steps of the noun technique?**

* Using the use cases, actors and other information about the system including inputs and outputs.
* Using other information from existing systems, current procedures, and current reports or forms, add items or categories of information needed.

**12. What is an attribute, an identifier or key, and a**

**compound attribute?**

* Attribute - descriptive pieces of information about things or objects  
  Identifier
* Key - an attribute the value of which uniquely identifies an individual thing or object
* Compound attribute - an attribute that consists of multiple pieces of information but its best treated in the aggregate

**13. What is an association, and what system development standard defines it?**

* A term, in UML, that describes a naturally occurring relationship between specific things, sometimes called a relationship.

**14. How would you describe or name the association**

**between a ship and a captain?**

* A ship has a captain, a captain has a ship.

**15. What is the term used for association by traditional analysts and database analysts?**

* Relationship.

**16. What is an association class? Why is an associa-**

**tion class used for modeling?**

* An association class is a complex data type that refines the relationship between two classes.

**17. What is multiplicity, and what is the otherterm used by traditional analysts and database analysts?**

* It is a measure of the number of links in an association between an object in one class and the objects in another class. In traditional analysis it is called cardinality.

**18. What is the minimum multiplicity for the association that reads a customer places zero or more orders?**

* Zero

**19. What is the maximum multiplicity for the association that reads an order is placed by exactly onecustomer?**

* One

**20. What are some examples of multiplicity constraints?**

* Customer places one or more order.
* Customer has only one account.
* Order is placed by only one customer.

**21. What are the three types of associations, and which is the most commonly used?**

* Binary, unary, ternary. Binary is most common.

**22. What are the three key parts of an entity-relationship**

**diagram (ERD)?**

* Data entities, relationships, and cardinality constraints.

**23. Sketch a simple ERD that shows a team has zero**

**or more players and each player is on one and**

**only one team.**

**24. Sketch a semantic net that shows two teams and**

**five players based on your ERD from question 23.**

**25. What is a class, a domain class, and the key parts**

**of a class diagram?**

* A class is a set of objects that are similar in nature and have the same "classification".
* A domain class is a class in the problem domain.
* A class diagram has classes with attributes, associations, and multiplicity constraints.

**26. What does a domain model class diagram show about system requirements, and how is it different from an ERD?**

* A domain model shows the classes (the things) and their relationships and constraints. These are the specific system requirements that must be built into the database. The problem domain classes are the classes from the domain and are "persistent", e.g. they must be stored in a database.
* An ERD has different notation than a domain model. An ERD is not as powerful as a domain model to model specific real world conditions.

**27. List appropriate UML class names by using the camelback notation for the following classes: graduate student, undergraduate major, course instructor, and final exam feedback.**

* GraduateStudent
* UndergraduateMajor
* CourseInstructor
* FinalExamFeedback

**28. List appropriate UML attribute names for the following attributes: student name, course grade, major name, and final exam quantity score.**

* studentName
* courseGrade
* majorName
* finalExamQuantityScore

**29. Draw a simple domain model class diagram for**

**the example in question #23 where a team has**

**zero or more players and each player is on one**

**and only one team.**

**30. Extend the domain model class diagram for**

**teams and players to show a record of game**

**statistics for each player in each game using an**

**association class.**

**31. In UML, what are three types of relationships**

**found on a class diagram?**

* **Regular association**
* **Generalization/Specialization**
* **Whole-part**

**32. What is a generalization/specialization relationship, and what object-oriented terms does it illustrate?**

* A generalization/specialization is a hierarchical relationship between classes, where some classes are subsets of other classes.

**33. Compare/contrast superclass and subclass. Com-**

**pare/contrasts abstract class and concrete class.**

* A superclass is higher in the relationship hierarchy and is a superset. A subclass is lower and is a subset of the superset. An abstract is a class with no objects allowed. It serves only as a template for attributes so that subclasses, which are concrete classes, will have objects with inherited attributes.

**34. What is a whole-part relationship, and why does**

**it show multiplicity?**

* A whole-part relationship in which a class is part of another class. It can have multiplicity constraints that allow multiple part-classes to belong to a single whole class.

**35. Compare/contrast aggregation with composition for a whole-part relationship.**

* Aggregation is where the "part" objects may exist outside of the whole-part relationship. This often applies to physical devices that can exist prior to becoming part of the aggregate. Composition is where the parts do not exist outside of the whole-part relationship. An example might be a sale which is "composed" of sale items. The sale items do not exist separate and apart from the "sale" object.

**36. What is an object state?**

* A condition during an object's life when it satisfies some criterion, performs some action, or waits for an event

**37. What is a state transition?**

* The movement of an object from one state to another state

**38. When considering requirements, states and state transitions are important for understanding which other diagram?**

* State machine diagram for an object class. Therefore, they help to understand the objects in the class diagram.

**39. What UML diagram is used to show the states**

**and transitions for an object?**

* State machine diagram

**40. List the elements that make up a transition description. Which elements are optional?**

* The elements are transition-name, guard-condition, and action-expression.
* All of these elements are optional. If a transition has none of these elements, it "fires" automatically when the object has finished any activity within the origin state.

**41. What is a composite state? What is it used for?**

* A composite state is a state containing multiple states and transitions.
* It represents a higher level of abstraction and can contain nested states and transition paths.

**42. What is meant by the term path?**

* A path is a sequential set of connected states and transitions.

**43. What is the purpose of a guard-condition?**

* A true/false test to see whether a transition can fire.