Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

1. Which of the following best defines composition in Java?
    a. A weak relationship where one object can exist independently.
    b. A strong relationship where two objects are dependent on each other.
    c. A "has-a" relationship where both objects are independent of each other.
    d. A relationship where one object inherits from multiple other objects.

Ans: b. A strong relationship where two objects are dependent on each other.

Explanation: Composition is a strong relationship where one object depends on another, and the contained object's lifetime is tied to the container.

2. What happens to the composed object when the container object is deleted in composition?
    a. The composed object can continue to exist.
    b. The composed object is also deleted.
    c. The composed object becomes independent.
    d. The composed object is transferred to another container.

Ans: b. The composed object is also deleted.

Explanation: In composition, when the container object is deleted, the composed object is also deleted because their lifetimes are tied together.

3. What is true about composition?
    a. Objects are loosely connected and can exist independently.
    b. The composed object cannot exist without the container object.
    c. The relationship is described as "uses-a" rather than "part-of."
    d. Both objects can exist separately and independently.

Ans: b. The composed object cannot exist without the container object.

Explanation: In composition, the composed object cannot exist without the container object, as their lifetimes are tightly bound together.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

4. How is composition different from aggregation?
    a. Composition creates a stronger dependency where the part cannot exist without the whole, while aggregation allows parts to live independently.
    b. Aggregation is stronger than composition.
    c. Composition allows multiple inheritance, while aggregation does not.
    d. Aggregation and composition are the same.

Ans: a. Composition creates a stronger dependency where the part cannot exist without the whole, while aggregation allows parts to live independently.

Explanation: Composition creates a stronger dependency where the part cannot exist without the whole, while aggregation allows parts to live independently.

5. When is composition used?
    a. When objects are independent and can exist separately.
    b. When one object contains another that cannot exist without it.
    c. When an object only uses methods from another, not sharing properties.
    d. When objects should exist independently but are loosely connected.

Ans: b. When one object contains another that cannot exist without it.

Explanation: Composition is typically used when one object contains another that cannot exist without it, indicating a strong relationship where the part depends on the whole.

6. What type of relationship does composition represent?
    a. "Has-a" relationship.
    b. "Part-of" relationship.
    c. "Uses-a" relationship.
    d. "Belongs-to" relationship.

Ans : b. "Part-of" relationship.

Explanation: Composition represents a "part-of" relationship, where one object is a part of another and cannot exist without it.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

7. What is the key characteristic of composition in Java?
    a. The child object can exist independently of the parent object.
    b. The child object cannot exist without the parent object.
    c. The parent object can exist without the child object.
    d. Composition allows for multiple inheritance in Java.

Ans: b. The child object cannot exist without the parent object.

Explanation: In composition, the child object cannot exist without the parent object.

8. In a composition relationship, what happens to the composed object if the container object is deleted?
    a. The composed object continues to exist independently.
    b. The composed object is also deleted.
    c. The composed object is transferred to another container object.
    d. The composed object becomes independent and survives.

Ans: b. The composed object is also deleted.

Explanation: In composition, when the container object is deleted, the composed object is also deleted because they are tightly bound together.

9. Which example demonstrates composition in Java?
    a. A person and their car, where the car can exist without the person.
    b. A human and their heart, where the heart cannot exist without the human.
    c. A library and books, where the books exist independently of the library.
    d. A teacher and students, where the students exist independently of the teacher.

Ans: b. A human and their heart, where the heart cannot exist without the human.

Explanation: Composition is demonstrated in the example of a human and their heart, as the heart cannot exist without the human. This shows a strong dependency between the two.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

10. Which statement is true about the composition relationship?
    a. Composition represents a weak association between classes.
    b. In composition, both objects can exist independently of each other.
    c. Composition represents a "part-of" relationship with strong dependency.
    d. Composition is a form of inheritance in Java.

Ans: c. Composition represents a "part-of" relationship with strong dependency.

Explanation: Composition represents a "part-of" relationship with a strong dependency, where the part cannot exist without the whole.

11. What is the main difference between aggregation and composition in Java?
    a. Aggregation represents a stronger relationship than composition.
    b. In aggregation, the child object can exist independently, while in composition, the child object depends on the parent object.
    c. Composition allows multiple inheritance, while aggregation does not.
    d. Aggregation and composition are the same in Java.

Ans: b. In aggregation, the child object can exist independently, while in composition, the child object depends on the parent object.

Explanation: The main difference is that in aggregation, the child object can exist independently, while in composition, the child object depends on the parent object and cannot exist without it.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

12. Which of the following is a feature of composition in Java?
    a. A composed object can exist without the parent object.
    b. Composition implies that the life cycle of the child object is dependent on the parent object.
    c. Composition uses the "has-a" relationship.
    d. Composition allows a child object to exist in multiple parent objects simultaneously.

Ans: b. Composition implies that the life cycle of the child object is dependent on the parent object.

Explanation: Composition implies that the life cycle of the child object is dependent on the parent object, meaning if the parent is deleted, the child is also deleted.

13. In which scenario would composition be the most appropriate relationship to use?
    a. A car and its engine, where the engine cannot function independently of the car.
    b. A company and its employees, where employees can exist independently of the company.
    c. A person and their house, where the house can exist without the person.
    d. A library and books, where books can exist without the library.

Ans: a. A car and its engine, where the engine cannot function independently of the car.

Explanation: Composition is most appropriate in the scenario where a car and its engine are involved, as the engine cannot function independently of the car.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

14. Which of the following best describes the relationship between objects in composition?
    a. The parent object and the child object are loosely coupled.
    b. The child object cannot exist without the parent object.
    c. The child object exists independently of the parent object.
    d. The child object can be part of multiple parent objects.

Ans: b. The child object cannot exist without the parent object.

Explanation: In composition, the child object cannot exist without the parent object, as their lifetimes are tightly bound together.

15. Which statement is true about the life cycle of objects in a composition relationship?
    a. The parent object is dependent on the child object.
    b. The parent object can exist without the child object, but the child object cannot live without the parent object.
    c. Both the parent and child objects can live independently.
    d. The life cycle of both objects is independent of each other.

Ans: b. The parent object can exist without the child object, but the child object cannot live without the parent object.

Explanation: In composition, the parent object can exist without the child object, but the child object cannot live without the parent object.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

16. In composition, how is the child object typically created?
    a. The child object is created separately and assigned to the parent object.
    b. The parent object creates the child object internally, and the child object cannot exist outside the parent.
    c. The child object is created by another external class and added to the parent.
    d. The child object is created using inheritance from the parent.

Ans: b. The parent object creates the child object internally, and the child object cannot exist outside the parent.

Explanation: In composition, the parent object creates the child object internally, and the child object cannot exist outside the parent.

17. Which of the following would be an incorrect use of composition in Java?
    a. A human and their heart, where the heart cannot exist without the human.
    b. A car and its wheels, where the wheels cannot function without the car.
    c. A school and its teachers, where the teachers can exist independently of the school.
    d. A house and its rooms, where rooms do not exist independently of the house.

Ans: c. A school and its teachers, where the teachers can exist independently of the school.

Explanation: In composition, the child object cannot exist independently of the parent. Therefore, a school and its teachers would be an incorrect use, as teachers can exist independently of the school.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

18. In composition, which of the following statements is false?
    a. The composed objects are usually created and destroyed together.
    b. Composition implies a strong "part-of" relationship.
    c. A composed object can be shared by multiple parent objects.
    d. Composition results in tight coupling between the parent and child objects.

Ans: c. A composed object can be shared by multiple parent objects.

Explanation: In composition, a composed object cannot be shared by multiple parent objects. It is tightly bound to the parent object, meaning the child object is exclusive to one parent.

19. Which keyword is often used in Java to implement composition when referencing another object?
    a. extends
    b. implements
    c. this
    d. No specific keyword; the parent object simply contains a reference to the child object.

Ans: d. No specific keyword; the parent object simply contains a reference to the child object.

Explanation: In composition, there is no specific keyword required; the parent object simply contains a reference to the child object, typically using an instance variable.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

20. How does composition affect code reusability?
    a. Composition generally leads to higher reusability as it allows independent usage of objects.
    b. Composition decreases reusability since the child objects are tightly coupled with the parent.
    c. Composition encourages code duplication rather than reusability.
    d. Composition does not affect code reusability at all.

Ans: a. Composition generally leads to higher reusability as it allows independent usage of objects.

Explanation: In composition, the tight coupling between the parent and child object can reduce reusability because the child objects are often bound to the specific parent object.

21. Which of the following relationships is best suited for composition?
    a. A class and its methods, where methods can exist independently of the class.
    b. A vehicle and its wheels, where wheels cannot function independently of the vehicle.
    c. A company and its employees, where employees can exist without the company.
    d. A building and its tenants, where tenants can exist independently of the building.

Ans: b. A vehicle and its wheels, where wheels cannot function independently of the vehicle.

Explanation: In composition, the relationship between a vehicle and its wheels is best suited, as the wheels cannot function independently of the vehicle, indicating a strong "part-of" relationship.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

22. Which of the following is NOT true about composition?
    a. Composition defines a "part-of" relationship.
    b. The composed object cannot exist without the parent object.
    c. Composition leads to a strong coupling between objects.
    d. The child object can exist independently outside the parent object.

Ans: d. The child object can exist independently outside the parent object.

Explanation: In composition, the child object cannot exist independently outside the parent object. Therefore, option d is not true.

23. What is the main advantage of composition over inheritance in Java?
    a. Composition allows for multiple inheritance, while inheritance does not.
    b. Composition enables a more flexible design and avoids tight coupling between classes.
    c. Composition is easier to implement than inheritance.
    d. Composition is faster than inheritance during program execution.

Ans: b. Composition enables a more flexible design and avoids tight coupling between classes.

Explanation: Composition enables a more flexible design and avoids tight coupling between classes, making it a preferred choice for creating loosely coupled systems.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

24. In a composition relationship, what is true about the ownership of the child object?
    a. The child object is owned by multiple parent objects.
    b. The parent object owns the child object and is responsible for its creation and destruction.
    c. The child object is owned by the child itself and can exist independently.
    d. There is no ownership relationship in composition.

Ans: b. The parent object owns the child object and is responsible for its creation and destruction.

Explanation: In composition, the parent object owns the child object and is responsible for its creation and destruction.

25. Which of the following is an example of a "part-of" relationship in Java?
    a. A university and its professors (professors can exist without the university).
    b. A person and their car (the car can exist independently of the person).
    c. A body and its heart (the heart cannot exist without the body).
    d. A customer and their bank account (the account can exist independently).

Ans: c. A body and its heart (the heart cannot exist without the body).

Explanation: A body and its heart is a "part-of" relationship, where the heart cannot exist without the body, indicating a strong composition relationship.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

26. Which of the following scenarios would NOT be suitable for composition?
    a. A computer and its motherboard (motherboard cannot exist without the computer).
    b. A school and its classrooms (classrooms can exist independently of the school).
    c. A house and its rooms (rooms cannot exist independently of the house).
    d. A car and its engine (engine cannot function without the car).

Ans: b. A school and its classrooms (classrooms can exist independently of the school).

Explanation: A school and its classrooms would not be suitable for composition, because classrooms can exist independently of the school, making it a better fit for aggregation rather than composition.

27. What does the term "tight coupling" in the context of composition mean?
    a. The objects in composition are loosely connected and can be used independently.
    b. The objects are highly dependent on each other, meaning if one is destroyed, the other is also destroyed.
    c. Composition allows for high flexibility between objects.
    d. There is no connection between the parent and child objects.

Ans: b. The objects are highly dependent on each other, meaning if one is destroyed, the other is also destroyed.

Explanation: In the context of composition, tight coupling means that the objects are highly dependent on each other. If the parent object is destroyed, the child object is also destroyed, indicating a strong dependency between them.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

28. How is composition typically used in Java to represent a "part-of" relationship?
    a. The parent object creates the child object and controls its life cycle.
    b. The child object is created externally and then assigned to the parent object.
    c. The parent and child objects do not interact with each other.
    d. Composition is not commonly used to represent "part-of" relationships in Java.

Ans: a. The parent object creates the child object and controls its life cycle.

Explanation: In composition, the parent object creates the child object and controls its life cycle, meaning the child object cannot exist independently of the parent.

29. When would you prefer composition over inheritance?
    a. When you need to model a "has-a" relationship.
    b. When you want to enable multiple inheritance.
    c. When you need to tightly couple the classes.
    d. When you want to use classes from different hierarchies together.

Ans: a. When you need to model a "has-a" relationship.

Explanation: You would prefer composition over inheritance when you need to model a "has-a" relationship, as it allows objects to be composed of other objects, providing more flexibility and loose coupling compared to inheritance.

Name of the Bundle	Proficient Bundle V2	Subject	Interview Skills in Programming V2
Topic	Composition	Last updated on	21 March 2025

30. Which form of association is stronger in terms of object dependency?

a. Aggregation
b. Composition
c. Inheritance
d. Association

Ans: b. Composition

Explanation: Composition is a stronger form of association because the contained object cannot exist without the container object. If the container is destroyed, the contained object is also destroyed.

31. Which relationship describes a "part-of" relationship in Java?

a. Aggregation
b. Inheritance
c. Composition
d. Association

Ans: c. Composition

Explanation: Composition is a "part-of" relationship where one class is a part of another and can't exist alone.