



Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Encapsulation	Last updated on	09 January 2025

1. What is Encapsulation?

- Hiding the class name
- Wrapping code and data together
- Making the class public
- Using multiple classes

Ans: b. Wrapping code and data together

Explanation: Encapsulation is the process of wrapping code and data together into a single unit (class).

2. What does encapsulation hide?

- Class methods
- Internal workings of a class
- Class variables
- Public methods

Ans: b. Internal workings of a class

Explanation: Encapsulation hides the internal workings of a class, such as how it performs tasks.

3. What is exposed in encapsulation?

- Internal variables
- Private methods
- Public interface
- Implementation details

Ans: c. Public interface

Explanation: In encapsulation, only the public interface is exposed to interact with the class.



Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Encapsulation	Last updated on	09 January 2025

4. What does encapsulation help achieve?

- a. Expose all class details
- b. Hide implementation details
- c. Make methods public
- d. Use static members

Ans: b. Hide implementation details

Explanation: Encapsulation helps hide the implementation details and only exposes necessary functionality.

5. How should instance variables be declared?

- a. public
- b. private
- c. protected
- d. default

Ans: b. private

Explanation: Instance variables should be private to keep them safe from outside access.

6. Where can private variables be used?

- a. Any class
- b. Only inside the same class
- c. Subclasses
- d. Same package

Ans: b. Only inside the same class

Explanation: Private variables can only be used inside the class where they are created.



Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Encapsulation	Last updated on	09 January 2025

7. What are getters and setters?

- Public methods to access private variables
- Private methods
- For storing data
- For creating objects

Ans: a. Public methods to access private variables

Explanation: Getters and setters help access and change private variables safely.

8. What does a setter do?

- Get the value
- Change the value
- Delete the value
- Create a value

Ans: b. Change the value

Explanation: A setter changes the value of a private variable.

9. What does a getter do?

- Get the value
- Change the value
- Delete the value
- Create a value

Ans: a. Get the value

Explanation: A getter retrieves the value of a private variable.



Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Encapsulation	Last updated on	09 January 2025

10. What is the purpose of getters and setters?

- a. To delete data
- b. To protect and manage data
- c. To create objects
- d. To print data

Ans: b. To protect and manage data

Explanation: Getters and setters help protect and control access to data in a class.

11. How do setters help with data?

- a. Check incoming data
- b. Hide the class name
- c. Make variables public
- d. Print values

Ans: a. Check incoming data

Explanation: Setters can check the incoming data before setting it to a variable.

12. Why should the internal state of a class be kept consistent?

- a. To prevent errors
- b. To make the code longer
- c. To allow direct access to data
- d. To avoid using getters and setters

Ans: a. To prevent errors

Explanation: Keeping the class state consistent helps avoid errors and unexpected behavior.



Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Encapsulation	Last updated on	09 January 2025

13. How do getters and setters help with data validation?

- a. By skipping data checks
- b. By enforcing data validation rules
- c. By deleting variables
- d. By making all data public

Ans: b. By enforcing data validation rules

Explanation: Getters and setters can enforce rules like checking if the data is valid before saving it.

14. What is an example of data validation using setters?

- a. Age should always be a positive number
- b. Age can be any value
- c. Age should be a string
- d. No restrictions on age

Ans: a. Age should always be a positive number

Explanation: A setter can ensure that values like age are always positive and valid.

15. Where can private variables be accessed?

- a. Inside the same class only
- b. From any class
- c. From subclasses
- d. From the whole package

Ans: a. Inside the same class only

Explanation: Private variables can only be used inside the class where they are declared.



Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Encapsulation	Last updated on	09 January 2025

16. Can other classes access private variables directly?

- a. Yes
- b. No
- c. Only in the same package
- d. Only in subclasses

Ans: b. No

Explanation: Private variables can only be accessed within the same class where they are declared.

17. How can private variables be accessed from outside the class?

- a. By making them public
- b. Using get() and set() methods
- c. By deleting the variable
- d. By creating a new variable

Ans: b. Using get() and set() methods

Explanation: Public get() and set() methods help access private variables safely.

18. What does the get() method do?

- a. Set a value
- b. Return the variable value
- c. Delete the variable
- d. Change the type

Ans: b. Return the variable value

Explanation: The get() method is used to return (fetch) the value of a private variable.



Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Encapsulation	Last updated on	09 January 2025

19. What does the set() method do?

- a. Return the value
- b. Set the value
- c. Delete the variable
- d. Hide the variable

Ans: b. Set the value

Explanation: The set() method is used to update or change the value of a private variable.

20. What will be the output if the following code is run?

```
public class Person {  
    private String name;  
    public String getName() { // Getter  
        return name;  
    }  
    public void setName(String newName) { // Setter  
        this.name = newName;  
    }  
}
```

- a. No output
- b. It will print null
- c. It will print name
- d. It will show an error

Ans: a. No output

Explanation: The class doesn't produce any output because no method is called in the main method.



Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Encapsulation	Last updated on	09 January 2025

21. What does using only the get() method for a class attribute make it?

- a. Read-only
- b. Write-only
- c. Read and write
- d. Read and delete

Ans: a. Read-only

Explanation: If only the get() method is used, the attribute can be read but not modified, making it read-only.

22. What does using only the set() method for a class attribute make it?

- a. Read-only
- b. Write-only
- c. Read and write
- d. Hidden

Ans: b. Write-only

Explanation: If only the set() method is used, the attribute can be modified but not accessed, making it write-only.

23. What is an advantage of using getters and setters?

- a. Easier to read
- b. Allows changes without affecting other parts
- c. Makes the code longer
- d. Prevents inheritance

Ans: b. Allows changes without affecting other parts

Explanation: Using getters and setters helps keep the internal structure of a class hidden, allowing changes without breaking other parts of the program.



Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Encapsulation	Last updated on	09 January 2025

24. What does making an attribute read-only mean?

- a. Can be changed but not read
- b. Can be read but not modified
- c. Can be accessed by any class
- d. Is hidden

Ans: b. Can be read but not modified

Explanation: Making an attribute read-only means it can be accessed, but not modified. This is done by only providing a get() method and no set() method.

25. What is an advantage of data hiding?

- a. Easy to test
- b. Better security
- c. Easy to create
- d. Faster development

Ans: b. Better security

Explanation: Data hiding improves security by preventing unauthorized access to internal details.

26. What does data hiding provide?

- a. Control over data
- b. Easier creation
- c. More variables
- d. More methods

Ans: a. Control over data

Explanation: Data hiding lets the class control how its data is accessed and modified.



Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Encapsulation	Last updated on	09 January 2025

27. What is an advantage of encapsulation?

- a. Easy to test
- b. Increased complexity
- c. Reduces code
- d. Makes data public

Ans: a. Easy to test

Explanation: Encapsulation organizes data and methods, making it easier to test.

28. What is an advantage of encapsulation?

- a. Better security
- b. No control
- c. No testing
- d. Slower development

Ans: a. Better security

Explanation: Encapsulation hides internal details, which increases security.

29. Why is encapsulation easy to create?

- a. Fewer classes
- b. Organizes data and methods
- c. Avoids writing code
- d. Hides everything

Ans: b. Organizes data and methods

Explanation: Encapsulation makes code easier to manage by grouping data and methods together.



Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Encapsulation	Last updated on	09 January 2025

30. What does encapsulation help control?

- a. Making code slower
- b. Making data public
- c. Data access
- d. Creating more classes

Ans: c. Data access

Explanation: Encapsulation gives control over how data is accessed and modified.

31. How should instance variables be declared to keep them safe?

- a. Class
- b. Private
- c. Protected
- d. Default

Ans: b.Private

Explanation: Using private keeps instance variables safe by restricting direct access from outside the class.

32. Which of the following is an advantage of encapsulation?

- a. Easy to test
- b. Increased complexity
- c. Reduces code
- d. Makes data public

Ans: a. Easy to test

Explanation: Encapsulation helps in testing because it keeps data safe and allows controlled access through methods, making it easier to verify and debug specific behaviors.



Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Encapsulation	Last updated on	09 January 2025

33. Which of the following is an advantage of encapsulation?

- a. Better security
- b. No control
- c. No testing
- d. Slower development

Ans: a. Better security

Explanation: An advantage of encapsulation is better security, as it hides the internal details of an object and only exposes necessary methods to interact with it.