

Encapsulation in DSA

1. What is Encapsulation?

- Encapsulation bundles **data** (variables) and **methods** (functions) that operate on the data into one unit (like a class).
 - It restricts direct access to data, allowing only controlled access through methods.
-

2. Why Encapsulation?

1. **Code Organization:** Groups data and operations in a logical structure.
 2. **Data Protection:** Prevents accidental or unauthorized modifications.
 3. **Reusability:** Makes code modular and reusable.
 4. **Ease of Maintenance:** Clear separation of concerns makes debugging easier.
-

3. How to Implement Encapsulation?

1. Use **private** access specifier for data members.
 2. Provide **public methods** (getters and setters) for controlled access.
 3. Hide unnecessary implementation details, exposing only required functionality.
-

4. Example: Encapsulation in Stack

Structure

- A Stack class with:
 - **Private:**
 - `arr[]` for storing elements,
 - `top` to track the top position,
 - `capacity` for the maximum size.
 - **Public:**
 - Methods like `push()`, `pop()`, `peek()`, and `isEmpty()` for operations.

Code:

5. Steps to Solve Encapsulation Problems

1. **Understand Requirements:**
 - What data to store?
 - What operations are needed?
2. **Design Class:**

- Use `private` for internal data (e.g., `arr`, `top`).
- Use `public` for interfaces (e.g., `push`, `pop`).

3. Handle Errors:

- Handle boundary cases like empty stack or full stack.

4. Restrict Access:

- Use getters/setters or public methods to control data access.
-

6. Key Points to Remember

- Encapsulation hides **implementation details** and exposes only **functionality**.
 - Use **access specifiers** (`private`, `public`) wisely.
 - Protect data integrity with controlled access.
-

7. Benefits of Encapsulation

1. **Security:** Prevents unauthorized data changes.
 2. **Modularity:** Code is easier to read and reuse.
 3. **Abstraction:** Hides complexity, focuses on what is necessary.
-

8. Example Use Cases

1. Stack
 2. Queue
 3. Linked List
-

<https://github.com/AjayPalhal-1/GrowthRepo>