1. What is Encapsulation in Java? Why is it called Data hiding?

   Encapsulation in Java is the concept of bundling the variables and methods that operate on the data into a single class. It restricts direct access to some of the object's components and can prevent unauthorized access and modification. This is why it is often referred to as Data Hiding because it hides the internal details of an object and only exposes necessary information through public methods.

2. What are the important features of Encapsulation?

   - Data Hiding: Internal object details are hidden from outside access.
   - Access Control: Provides control over how data is accessed and modified via getters and setters.
   - Code Maintenance: Makes it easier to maintain and modify code without affecting external code.
   - Security : Enhances the security as it provides very controlled access of its member variables.

3. What are getter and setter methods in Java Explain with an example

   Getter methods allow access to the value of a private variable, while setter methods allow modification of that value.

```
public class Student {
    private String name;
    private int age;

    // Getter for name
    public String getName() {
        return name;
    }

    // Setter for name
    public void setName(String name) {
        this.name = name;
    }

    // Getter for age
    public int getAge() {
        return age;
    }

    // Setter for age
    public void setAge(int age) {
        this.age = age;
    }

    public static void main(String[] args) {
        Student student = new Student();
        student.setName("John");
        student.setAge(20);

        System.out.println("Name: " + student.getName());
        System.out.println("Age: " + student.getAge());
    }
}
```

4. What is the use of this keyword explain with an example?

   The this keyword in Java refers to the current object of the class. It is mainly used to differentiate between instance variables and local variables when they have the same name, or to call the current object's methods.

```
public class Student {
    private String name;

    public void setName(String name) {
        this.name = name;  // "this" refers to the instance variable
    }
    public void display() {
        System.out.println("Name: " + this.name);
    }

    public static void main(String[] args) {
        Student student = new Student();
        student.setName("Alice");
        student.display();
    }
}
```

5. What is the advantage of Encapsulation?

- Control over data: You can control how the data is accessed or modified by using getter and setter methods.
- Improved security: By hiding the internal details, encapsulation protects the object's state from unauthorized access.
- Code maintainability: It makes the code more maintainable and flexible, allowing changes to the internal workings of a class without affecting external code.

6. How to achieve encapsulation in Java? Give an example.

To achieve encapsulation in Java, follow these steps:

- Declare the variables as private.
- Provide public getter and setter methods to access and update the values of the variables.

```
public class BankAccount {
    private double balance;

    // Getter method
    public double getBalance() {
        return balance;
    }

    // Setter method
    public void deposit(double amount) {
        if(amount > 0) {
            balance += amount;
        }
    }

    public static void main(String[] args) {
        BankAccount account = new BankAccount();
        account.deposit(500);
        System.out.println("Balance: " + account.getBalance());
    }
}
```