

Java – Инкапсуляция

Инкапсуляция — это одна из четырёх фундаментальных концепций ООП. Названия оставшихся трёх: [наследование](#), [полиморфизм](#), [абстракция](#).

Инкапсуляция в Java является механизмом обёртывания данных (переменных) и кода, работающего с данными (методами), в одно целое. В инкапсуляции переменные класса будут скрыты от других классов и доступ к ним может быть получен только с помощью метода их текущего класса. По-другому это называется скрытием данных.

Для достижения инкапсуляции в Java:

- Объявите переменные класса как `private`.
- Предоставьте `public` к методам установки и получения (сеттеру и геттеру) для изменения и просмотра значений переменных.

Пример

Ниже приведён пример процесса инкапсуляции в Java:

```
/* File name : EncapTest.java */
public class EncapTest {
    private String name;
    private String idNum;
    private int age;

    public int getAge() {
        return age;
    }

    public String getName() {
        return name;
    }

    public String getIdNum() {
        return idNum;
    }

    public void setAge(int newAge) {
        age = newAge;
    }

    public void setName(String newName) {
        name = newName;
    }

    public void setIdNum(String newId) {
        idNum = newId;
    }
}
```

Публичные методы `setXXX()` и `getXXX()` являются точками доступа переменных экземпляра класса `EncapTest`. Обычно эти методы упоминаются как геттеры и сеттеры. Следовательно, любой класс, который хочет получить доступ к переменным, должен получить к ним доступ через эти геттеры и сеттеры.

Доступ к переменным класса `EncapTest` можно получить следующей программой:

```
/* File name : RunEncap.java */
public class RunEncap {

    public static void main(String args[]) {
        EncapTest encap = new EncapTest();
        encap.setName("Олег");
        encap.setAge(28);
        encap.setIdNum("12343ms");

        System.out.print("Имя: " + encap.getName() + ", возраст: " + encap.getAge());
    }
}
```

Мы получим следующий результат:

Преимущества инкапсуляции

- Поля класса можно сделать только для чтения или только для записи.
- Класс может иметь полный контроль над тем, что хранится в его полях.