



Объектно-ориентированное программирование



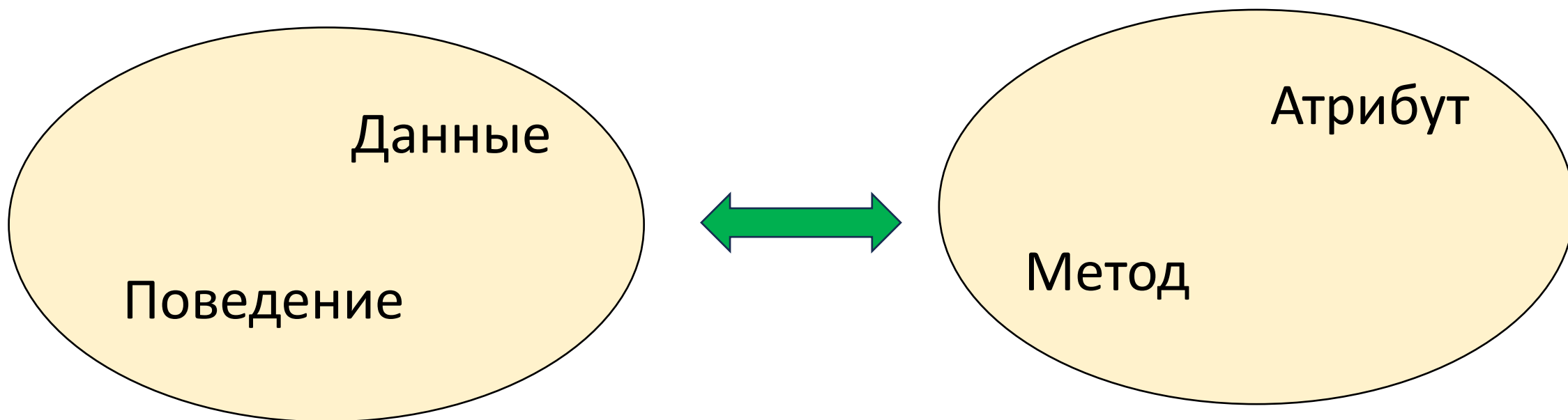
Объектно-ориентированное программирование

Объект - ?

Объект – сущность, одновременно
содержащая данные и поведение



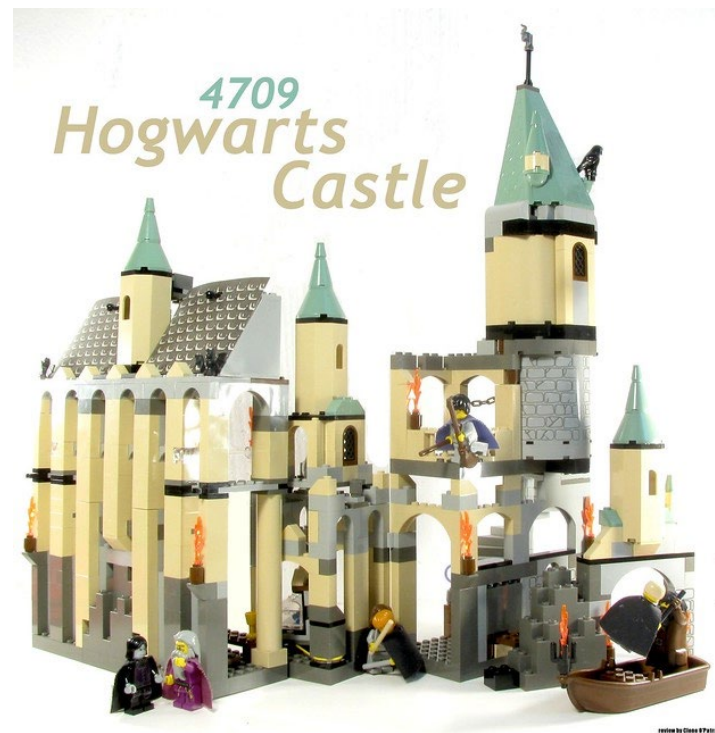
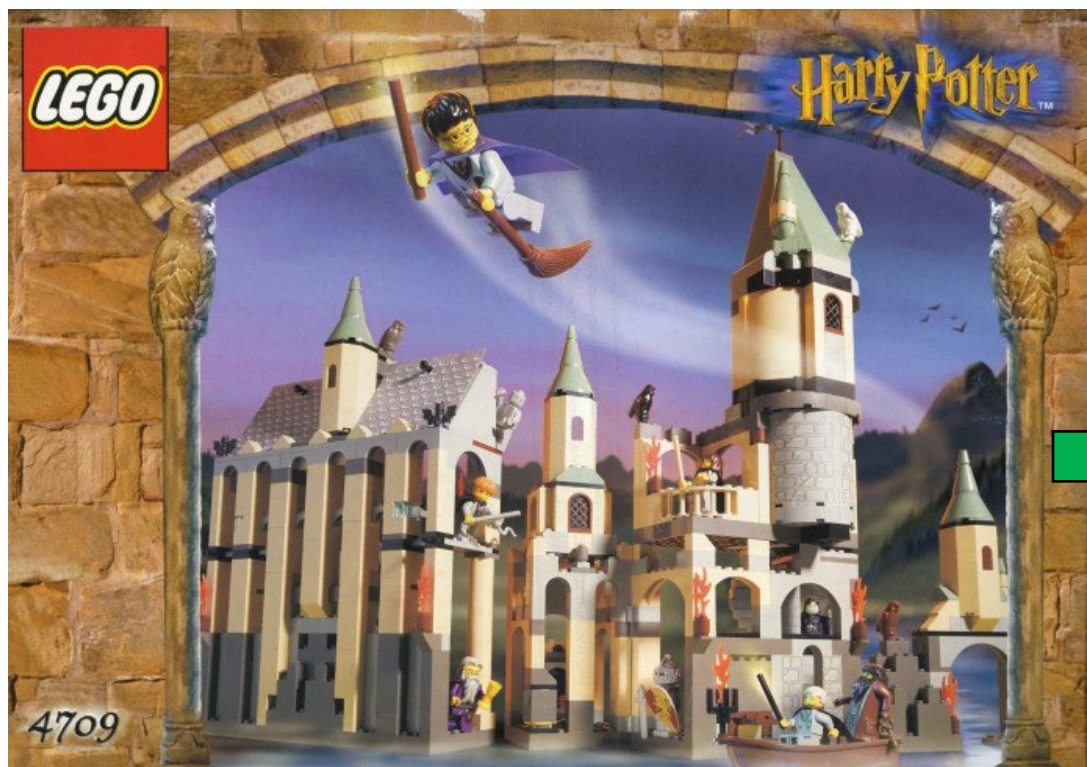
Объект





Класс

Класс – чертеж объекта





Три кита ООП

Объектно-ориентированное программирование



Наследование

Инкапсуляция

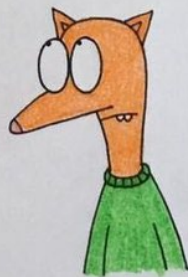
Полиморфизм





Инкапсуляция

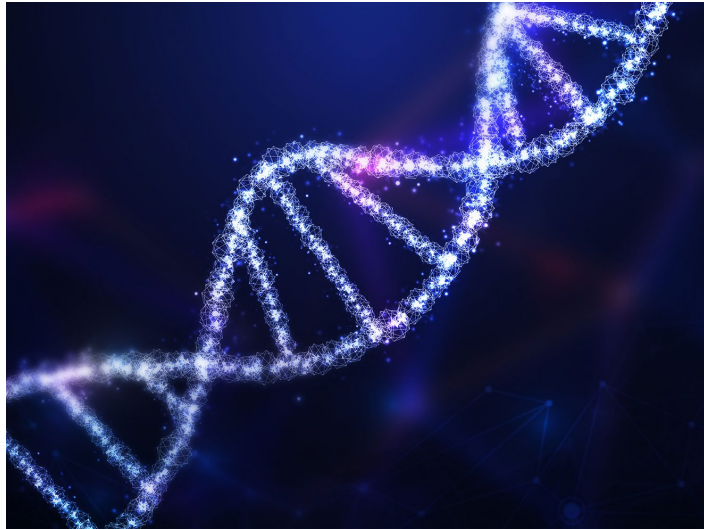
ЧТО ВНУТРИ
У КОЛОБКА



Инкапсуляция - это механизм, который объединяет данные и методы, манипулирующие этими данными, и защищает и то и другое от внешнего вмешательства или неправильного использования.



Наследование

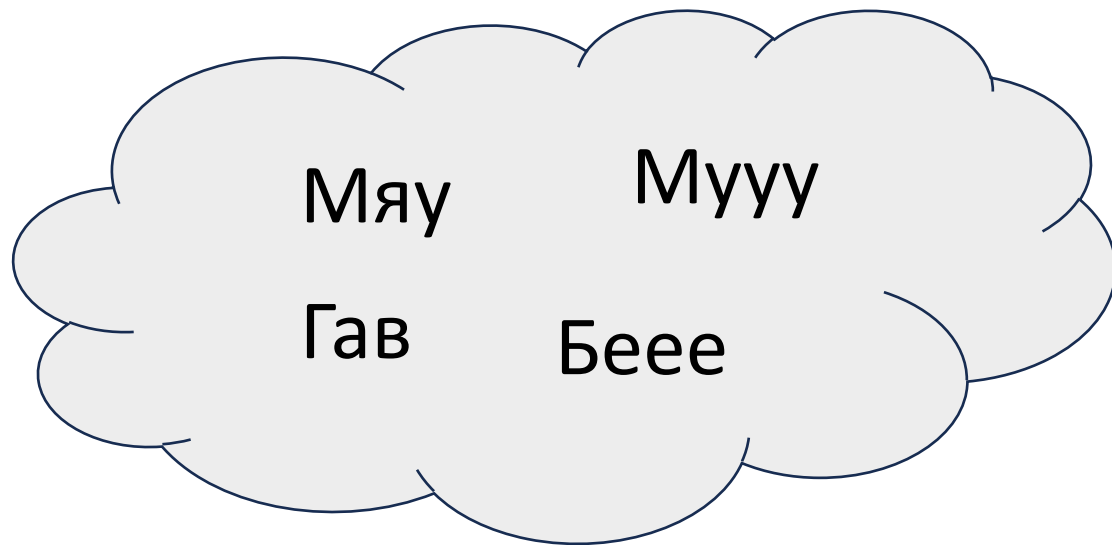


Наследование – это процесс, посредством которого, один объект может приобретать свойства другого. Точнее, объект может наследовать свойства другого объекта и добавлять к ним черты, характерные только для него.





Полиморфизм



Полиморфизм – это свойство, которое позволяет одно и тоже имя использовать для решения нескольких технически разных задач.



Обращение к элементам класса

```
#include <iostream>
```

```
class Point {  
public:  
    int x;  
    int y;  
};
```

```
int main()  
{
```

```
    Point P1, *P2;  
    P2 = new Point;  
    P1.x = 10;
```

```
    (*P2).x = 10;    //Скобки нужны, т.к. у точки более высокий приоритет, чем у *
```

```
    std::cout << P1.x << std::endl;
```

```
    std::cout << (*P2).x << std::endl;
```

```
    std::cout << P2->x << std::endl;    //правильный способ!
```

```
    delete P2;
```

```
}
```

Обращение к полям класса производится с помощью операторов . и ->

. – обращение к объекту напрямую

-> - при обращении к объекту по указателю

$(*P).x \iff P \rightarrow x$

Полный код см. в комментариях



Задания

Вариант 1: Банковский счёт

Создай класс `BankAccount` с приватными полями:

номер счёта

баланс

Методы:

`deposit(amount)` — пополнение

`withdraw(amount)` — снять, если хватает денег

`getBalance()` — узнать баланс

Добавить возможность перекладывать деньги с одного счёта на другой

Вариант 2 : Термометр

Класс `Thermometer` с приватным полем:

температура в градусах Цельсия

Методы:

`setTemperature(value)` — установить температуру (ограничить от -100 до +100)

`getTemperature()` — получить температуру

`toFahrenheit()` — вернуть температуру в Фаренгейтах

Добавить возможность синхронизировать значения термометров