

Классы и объекты

Класс описывает, какими должны быть объекты

Robot

- сила
- скорость
- броня

Идти()

Стрелять()

Классы и объекты

Robot

- сила
- скорость
- броня

Идти()
Стрелять()

Robot1



Robot2



Robot3



Определение класса

```
class Robot():  
    pass
```

Создание методов и данных в классе

```
class Robot():  
    power = 30  
    speed = 100  
    armor = 25  
  
    def walk(self):  
        print('иду')  
  
    def shoot(self):  
        print('стреляю')
```

Создание объектов

```
class Robot():
    power = 30
    speed = 100
    armor = 25

    def walk(self, time):
        distance = self.speed * time
        print('прошел %s метров' % distance)

    def shoot(self):
        print('стреляю')

robo1 = Robot()
robo1.speed = 140
robo2 = Robot()
time = 123
robo1.walk(time)
robo2.walk(time)
```

В функции можно передавать объекты

```
class Robot():  
    power = 30  
    speed = 100  
    armor = 25  
    name = 'Noname'  
  
    def walk(self, time):  
        distance = self.speed * time  
        print('прошел %s метров' % distance)  
  
    def shoot(self):  
        print('стреляю')
```

продолжение

```
class Arena():
    def fight(self, robo1, robo2):
        if robo1.power > robo2.power:
            print('Winner is %s' % robo1.name)
        elif robo1.power < robo2.power:
            print('Winner is %s' % robo2.name)
        else:
            print('Nobody wins')
```

```
r1 = Robot()
r1.power = 40
r1.name = 'Vasya'
r2 = Robot()
r2.name = 'Petya'
```

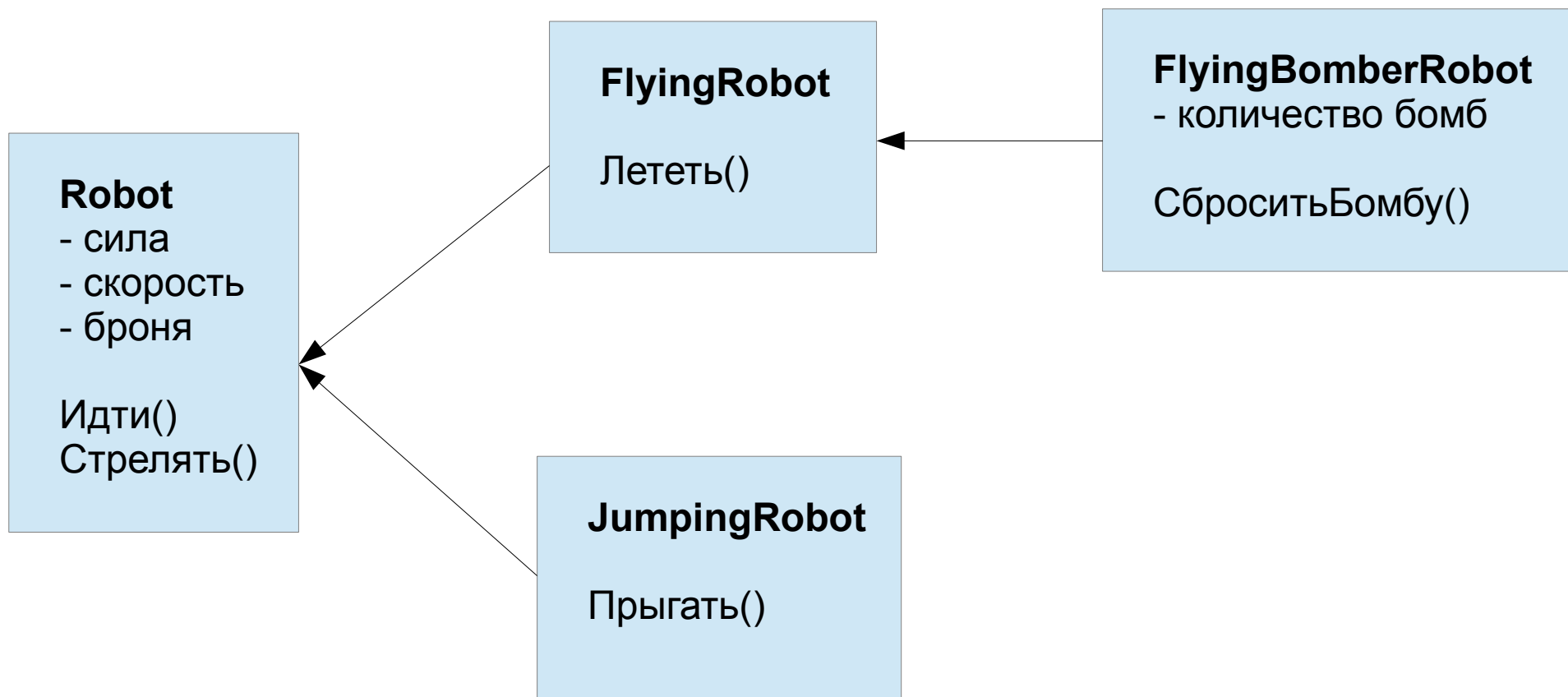
```
arena = Arena()
arena.fight(r1, r2)
```

Объекты и классы на примере Черепашки

```
import turtle
pen1 = turtle.Pen()
pen2 = turtle.Pen()
pen1.forward(50)
pen1.right(90)
pen1.forward(20)
pen2.left(90)
pen2.forward(100)
turtle.exitonclick()
#создайте третий объект Pen
```


Придумайте каждый свой
собственный класс со свойствами
и функциями и создайте
несколько его объектов. Напрмер,
автомобиль или дом или что
хотите

Наследование



Наследование

```
class A():  
    def a(self):  
        print('a')
```

```
class B(A):  
    def b(self):  
        print('b')
```

```
objectA = A()  
objectA.a()
```

```
objectB = B()  
objectB.a()  
objectB.b()
```

Вызов функций внутри классов

```
class Animal():  
  
    def move(self):  
        print('moving...')  
  
    def eat(self):  
        print('yum-yum-yum...')  
  
    def findFood(self):  
        self.move()  
        self.eat()  
  
animal = Animal()  
animal.findFood()
```

Инициализация объектов

```
class Man:
    def __init__(self, name):
        self.name = name

    def greeting(self):
        print("Hello, my name is %s" % self.name)

m1 = Man('Vasya')
m2 = Man('Petya')

m1.greeting()
m2.greeting()
```

Торт

Из прямоугольного торта вырезали прямоугольный кусок. Расположение и ориентация вырезанного куска произвольна. Как разделить торт одним прямолинейным разрезом? (Горизонтально резать нельзя).

