



C++

## Разбор домашней работы



# Классы

# Теория



15 минут



# Классы

**Класс** - разновидность абстрактного типа данных в объектно-ориентированном программировании.

**Для чего может понадобиться  
создать и использовать класс?**

# Объявление класса

```
class Pet {  
public:  
    string name;  
    string animal_type;  
    int age;  
};
```

# Объявление класса

```
class Pet {  
public:  
    string name;  
    string animal_type;  
    int age;  
};
```

**Важно!**





# Некоторые определения

**Класс** - абстрактный тип данных

**Объект** - экземпляр класса (переменная этого типа)

**Поле/свойство** - переменная, принадлежащая классу или объекту этого класса

# Объект класса

```
class Pet {  
public:  
    string name;  
    string animal_type;  
    int age;  
};  
int main() {  
    Pet p;  
    cin >> p.animal_type >> p.name >> p.age;  
}
```

# Объект класса

```
class Pet {  
public:  
    string name;  
    string animal_type;  
    int age;  
};  
  
int main() {  
    Pet p1, p2;  
    cin >> p1.animal_type >> p1.name >> p1.age;  
}
```

p1: name anima_type age	p2: name anima_type age
----------------------------------	----------------------------------

# Практика



20 минут



# Как передать объект класса в функцию

# Теория



10 минут



# Передача объектов в функции

Передача объектов классов в функции ничем не отличается от передачи любой другой переменной.

Если необходимо изменить объект внутри функции, его необходимо передать по **ссылке**.

# Примеры

```
class Pet {  
public:  
    string name;  
    string animal_type;  
    int age;  
};  
void print_pet(Pet p) {  
    cout << p.animal_type << " " << p.name << " " << p.age;  
}  
void read_pet(Pet& p) {  
    cin >> p.animal_type >> p.name >> p.age;  
}
```



# Практика



25 минут



# Итоги урока

- 1) Что такое класс
- 2) Что такое объект класса
- 3) Чем отличается класс от объекта
- 4) Как обращаться к отдельным полям класса
- 5) Как передавать объекты в функции