**Домашняя работа №4**

**Задание:**

1. Создать класс Animal. Добавить в класс приватное поле возраст(int) и публичное  поле название(string) животного.
2. Создать три конструктора для этого класса:

* без параметров ()
* с одним параметром (string name)
* с двумя параметрами (string name, int age)

В функции main создать пять объектов класса Animal.

1. Создать два метода compareAge (метод должен возвращать true или false)

* c параметром int age (сравнить возраст животного с числом)
* с параметром Animal a (сравнить возраст двух животных между собой)

\*Почему вместо параметра “Animal a” лучше написать “const Animal& a”?

**Код:**

#include <iostream>

#include <string>

#include <Windows.h>

using namespace std;

class Animal {

private:

int age;

public:

string name;

Animal() {}

Animal(string \_name): name(\_name) {}

Animal(string \_name, int \_age): name(\_name), age(\_age) {}

bool compareAge(int num) { return this->age == num ; }

//const юзаем, чтобы обезопасить и не изменить объект класса в этой функции.

// а & юзаем, чтобы не передавать копию, а только ссылку

//bybus or bob colbaskin?? 🤨🥸

bool compareAge(const Animal& a) { return this->age == a.age; }

};

int main() {

int num = 20;

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

Animal lion("Прав и Лев", -11);

Animal zebra;

Animal php("Наш слон");

Animal python("Змеюка", 43);

Animal bear("Пивкоо!", -11);

cout << "lion: " << boolalpha << lion.compareAge(num) << endl;

cout << "bear: " << boolalpha << bear.compareAge(num) << endl;

cout << "lion and bear: " << boolalpha << lion.compareAge(bear);

}

**Вывод:**

