

**Тема:** Составление программ с использованием ООП

**Цель:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с ООП

### ПЗ-17.1

**Постановка задачи:** Вариант 29

Создайте класс «Календарь», который имеет атрибуты год, месяц и день. Добавьте методы для определения дня недели, проверки на високосный год и определения кол-ва дней в месяце

**Текст программы:**

```
import datetime

class Calendar:
    def __init__(self, year, month, day):
        self.year = year
        self.month = month
        self.day = day

    def day_of_week(self):
        date_obj = datetime.date(self.year, self.month, self.day)
        return date_obj.strftime("%A")

    def is_leap_year(self):
        if self.year % 4 == 0:
            if self.year % 100 == 0:
                if self.year % 400 == 0:
                    return True
                else:
                    return False
            else:
                return True
        else:
            return False

    def days_in_month(self):
        if self.month == 2: # февраль
            if self.is_leap_year():
                return 29
            else:
                return 28
        elif self.month in [4, 6, 9, 11]: # апрель, июнь, сентябрь, октябрь
            return 30
        else:
            return 31

my_calendar = Calendar(2023, 4, 23)

print("День недели:", my_calendar.day_of_week())
print("Високосный год:", my_calendar.is_leap_year())
print("Количество дней в месяце:", my_calendar.days_in_month())
```

**Протокол программы:**

День недели: Sunday

Черячукин Кирилл ИС-21

Високосный год: False

Количество дней в месяце: 30

Process finished with exit code 0

## ПЗ-17.2

### Постановка задачи:

Создайте базовый класс «Животное» со свойствами «вид», «кол-во лап», «цвет шерсти» от этого класса унаследуйте класс «собака» и добавьте в него свойства « кличка» и «порода»

### Текст программы:

```
class Animal:
    def __init__(self, kind, num_paws, fur_color):
        self.kind = kind
        self.num_paws = num_paws
        self.fur_color = fur_color

class Dog(Animal):
    def __init__(self, kind, num_paws, fur_color, name, breed):
        super().__init__(kind, num_paws, fur_color)
        self.name = name
        self.breed = breed

animal = Animal("Мустанг", 4, "Коричневый")
print(animal.kind)
print(animal.num_paws)
print(animal.fur_color)

dog = Dog("Мустанг", 4, "Коричневый", "Форд", "Лабрадор")
print(dog.kind)
print(dog.num_paws)
print(dog.fur_color)
print(dog.name)
print(dog.breed)
```

### Протокол программы:

Мустанг  
4  
Коричневый  
Мустанг  
4  
Коричневый  
Форд  
Лабрадор

Process finished with exit code 0

### Вывод:

Я приобрел навыки составление программ с ООП