

Тема: закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрёл навыки составление программ с использованием ООП в IDE PyCharm Community.

Текст программы:

1)

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

# Для задачи из блока 1 создать две функции, save_def и load_def, которые
# позволяют
# сохранять информацию из экземпляров класса (3 шт.) в файл и загружать ее
# обратно.
# Использовать модуль pickle для сериализации и десериализации объектов Python в
# бинарном формате.
import
datetime
import pickle

class Calendar:
    def __init__(self,
year, month, day):
        self.year = year
self.month = month
self.day = day
    def
day_of_week(self):
        date = datetime.date(self.year, self.month, self.day)
days_of_week = ["Понедельник", "Вторник", "Среда", "Четверг", "Пятница",
"Суббота", "Воскресение"]
        return
days_of_week[date.weekday()]
    def
is_leap_year(self):
        return (self.year % 4 == 0 and self.year % 100 != 0) or (self.year %
400 == 0)
    def days_in_month(self):
        if self.month == 12:
            next_month = datetime.date(self.year + 1, 1, 1)
        else:
            next_month = datetime.date(self.year, self.month + 1, 1)
last_day_of_month = next_month - datetime.timedelta(days=1)
        return
last_day_of_month.day
    def save_def(filename,
instances):
```

```

    with open(filename, 'wb') as f:
pickle.dump(instances, f)

def load_def(filename):
    with
open(filename, 'rb') as f:
        return pickle.load(f)

year = int(input("Введите год: ")) month
= int(input("Введите месяц: ")) day =
int(input("Введите день: "))
date = Calendar(year, month,
day) save_def('dates.bin',
[date])
loaded_dates =
load_def('dates.bin')
for date in
loaded_dates:
    print(f'Дата: {date.year}-{date.month}-{date.day}')
print(f'День недели: {date.day_of_week()}')    print(f'Весокосный
год: {"Да" if date.is_leap_year() else "Нет"}')    print(f'Дней в
месяце: {date.days_in_month()}')

```

Протокол программы:

```

PS C:\Users\Elnur\Desktop\PZ> &
'c:\Users\Elnur\AppData\Local\Microsoft\WindowsApps\python3.11.exe'
'c:\Users\Elnur\.vscode\extensions\ms-python.debugpy-2024.6.0-
win32x64\bundled\libs\debugpy\adapter\..\debugpy\launcher' '50359' '--'
'C:\Users\Elnur\Desktop\PZ\PZ\PZ_16\PZ_16_1.py'

```

Введите год: 2006

Введите месяц: 11

Введите день: 28

Дата: 2006-11-28

День недели: Вторник

Весокосный год: Нет

Дней в месяце: 30

```
PS C:\Users\Elnur\Desktop\PZ>
```

Текст программы:

2)

```
# Создайте класс "Животное", который содержит информацию о виде и возрасте
# животного. Создайте классы "Собака" и "Кошка", которые наследуются от класса
# "Животное" и содержат информацию о породе.

class Animal:
    def __init__(self, species, age):
        self.species = species
        self.age = age

class Dog(Animal):
    def __init__(self, species, age, breed):
        super().__init__(species, age)
        self.breed = breed

class Cat(Animal):
    def __init__(self, species, age, breed):
        super().__init__(species, age)
        self.breed = breed

# Пример использования:
print("-----")
dog = Dog("Собака", 3, "Лабрадор")
print(dog.species)
print(dog.age)
print(dog.breed)
print("-----")

cat = Cat("Кошка", 5, "Персидская")
print(cat.species)
print(cat.age)
print(cat.breed)
print("-----")
```

Протокол программы:

```
PS C:\Users\Elnur\Desktop\PZ\PZ\PZ_16> &
'c:\Users\Elnur\AppData\Local\Microsoft\WindowsApps\python3.11.exe'
'c:\Users\Elnur\.vscode\extensions\ms-python.debugpy-2024.6.0-win32-
x64\bundled\libs\debugpy\adapter\..\debugpy\launcher' '59199' '--'
'C:\Users\Elnur\Desktop\PZ\PZ\PZ_16\PZ_16_2.py'
```

Собака

3

Лабрадор

Кошка

5

Персидская

PS C:\Users\Elnur\Desktop\PZ\PZ\PZ_16>

Вывод: в процессе выполнения практического занятия выработал навыки составления программ в IDE PyCharm Community. Выполнены разработка кода, отладка, тестирование, оптимизация программного кода.