



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»
РТУ МИРЭА

Институт информационных технологий

Кафедра математического обеспечения и стандартизации информационных
технологий

ОТЧЁТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №1
по дисциплине «Программирование на Python»

Тема: «Основы языка Python»

Выполнили студенты
группы

ИКБО-11-22

Иванов И. И.
Пупкин В. В.

Принял преподаватель

Иванов П. Ф.

Работа выполнена

«__» _____ 20__

(Подпись студента)

Зачтено

«__» _____ 20__

(Подпись преподавателя)

Москва 2025

Содержание

1. Ход работы	3
1.1. Введение	3
1.2. Пример работы с картинками	3
1.3. Пример работы с листингами	3
1.3.1. Простые листинги	3
1.3.2. Листинг с подсветкой	3
1.3.3. листинг из файла	3
1.3.4. Частичное цитирование файла	6
1.4. Таблицы	7

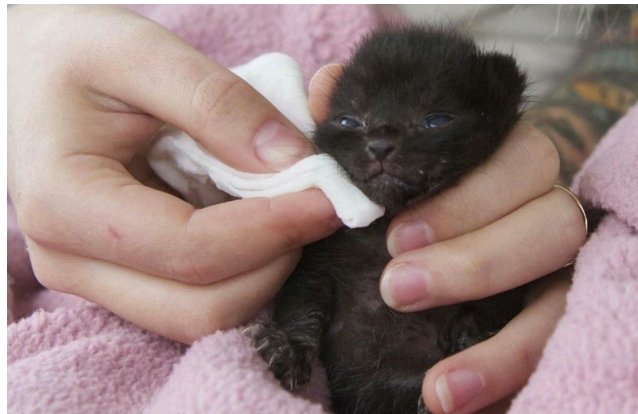
1. Ход работы

1.1. Введение

Этот шаблон подходит для любых мирэашных отчетов, в т.ч. курсовых. Если что-то работает неправильно, пишите в тг [@burenin-a](https://t.me/burenin-a)

1.2. Пример работы с картинками

С помощью функции `picture` можно добавить картинку с подписью в документ, таким образом, был добавлен Рисунок 1



Эта публикация отражает ваш стиль жизни? ✕



Однозначно нет Однозначно да

Рисунок 1 — Так называемый мем

1.3. Пример работы с листингами

1.3.1. Простые листинги

Для создания листинга можно использовать функцию `listing`

Листинг 1 — Программа выводит hello world в консоль

```
print('Hello, world!')
```

1.3.2. Листинг с подсветкой

Для создания листинга можно использовать функцию `listing`

Листинг 2 — Программа выводит hello world в консоль

```
print("Hello, world")
```

1.3.3. листинг из файла

Здесь можно увидеть как добавить листинг программы из файла

Листинг 3 — Некоторая программа на языке python

```
import abc
import math

from .errors import InvalidArgumentsError

class Shape(abc.ABC):
    """
    Represents base geometric shape
    """

    @abc.abstractmethod
    def square(self) -> float:
        """
        Calculates square of the shape

        :return: square of the shape
        """

class Circle(Shape):
    """
    Represents circle using its radius

    :argument radius: Radius of the circle
    :raises: InvalidArgumentsError
    """

    def __init__(self, radius: float) -> None:
        if radius <= 0:
            raise InvalidArgumentsError(
                f"Radius must be positive, but value {radius}
was provided"
            )

        self._radius = radius

    @property
    def radius(self) -> float:
        """
        Returns radius of the circle

        :return: radius of the circle
        """
        return self._radius

    def square(self) -> float:
        """
        Calculates square of the circle

        :return: Radius of the circle
        """
```

Продолжение листинга 3

```
        return self._radius**2 * math.pi

    def __repr__(self) -> str:
        return f"Circle(radius={self.radius})"

class Triangle(Shape):
    """
    Represents a triangle using its sides

    :param a: First side of the triangle
    :param b: Second side of the triangle
    :param c: Third side of the triangle
    :raises InvalidArgumentsError: if triangle with given sides does
not exist
    :returns: Square of triangle with given sides
    """

    def __init__(self, a: float, b: float, c: float) -> None:
        self.__check_constraints(a, b, c)
        self._a = float(a)
        self._b = float(b)
        self._c = float(c)

    @property
    def a(self) -> float:
        return self._a

    @property
    def b(self) -> float:
        return self._b

    @property
    def c(self) -> float:
        return self._c

    def with_sides(
        self,
        a: float | None = None,
        b: float | None = None,
        c: float | None = None,
    ) -> "Triangle":
        """
        Returns new triangles with updated sides

        :param a: New first side of the triangle
        :param b: New second side of the triangle
        :param c: New third side of the triangle
        :raises InvalidArgumentsError: if triangle with given sides
does not exist
        :return: triangle with updated sides
        """
        return Triangle(a or self.a, b or self.b, c or self.c)
```

Продолжение листинга 3

```
def is_right(self, places: int = 7) -> bool:
    """
    Check if triangle is right by checking the Pythagorean theorem.

    :param places: Precision of checking. 7 by default.
    :returns: true if triangle is right
    """

    # Check if Pythagorean theorem is satisfied
    c = max(self.a, self.b, self.c)
    if c == self.a:
        a, b = self.b, self.c
    elif c == self.b:
        a, b = self.a, self.c
    else:
        a, b = self.a, self.b

    return round(a * a + b * b, places) == round(c * c, places)

def square(self) -> float:
    """
    Calculates square of the triangle

    :return: square of the triangle
    """
    a, b, c = self.a, self.b, self.c
    p = (a + b + c) / 2
    return math.sqrt(p * (p - a) * (p - b) * (p - c))

def __repr__(self):
    return f"Triangle(a={self.a}, b={self.b}, c={self.c})"

@staticmethod
def __check_constraints(a: float, b: float, c: float) -> None:
    if a <= 0 or b <= 0 or c <= 0:
        raise InvalidArgumentsError(
            f"All sides must be positive, but ({a}, {b}, {c})
were given"
        )

    if not (a + b > c and a + c > b and b + c > a):
        raise InvalidArgumentsError(
            f"Triangle with sides ({a}, {b}, {c}) does not exists"
        )
```

1.3.4. Частичное цитирование файла

На листинге листинге 4 можно увидеть, каким образом вставить только часть содержимого файла

Листинг 4 — Абстрактная функция square

```
@abc.abstractmethod
def square(self) -> float:
    """

    Calculates square of the shape

    :return: square of the shape
    """
```

1.4. Таблицы

Для создания таблиц используется функция `mytable`, она позволяет создавать таблицы с правильными подписями и хедерами

Таблица 1 — Более сложная таблица

Ученик	Оценки		
	Русский язык	Математика	Литература
Иванов И.И.	86	90	54
Пупки В.А.	42	12	64
Афанасьев А.О.	86	80	76
Иванов И.И. 1	86	90	54
Иванов И.И. 2	86	90	54
Иванов И.И. 3	86	90	54
Иванов И.И. 4	86	90	54
Иванов И.И. 5	86	90	54
Иванов И.И. 6	86	90	54
Иванов И.И. 7	86	90	54
Иванов И.И. 8	86	90	54
Иванов И.И. 9	86	90	54
Иванов И.И. 10	86	90	54
Иванов И.И. 11	86	90	54
Иванов И.И. 12	86	90	54
Иванов И.И. 13	86	90	54
Иванов И.И. 14	86	90	54
Иванов И.И. 15	86	90	54
Иванов И.И. 16	86	90	54
Иванов И.И. 17	86	90	54
Иванов И.И. 18	86	90	54
Иванов И.И. 19	86	90	54

Продолжение таблицы 1

Ученик	Оценки		
	Русский язык	Математика	Литература
Иванов И.И. 20	86	90	54
Иванов И.И. 21	86	90	54
Иванов И.И. 22	86	90	54
Иванов И.И. 23	86	90	54
Иванов И.И. 24	86	90	54
Иванов И.И. 25	86	90	54
Иванов И.И. 26	86	90	54
Иванов И.И. 27	86	90	54
Иванов И.И. 28	86	90	54
Иванов И.И. 29	86	90	54