

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ»

Институт информатики, математики и робототехники  
Кафедра математического и компьютерного моделирования

**Лабораторная работа №1: Работа с форматом INI в Python**

**ОБУЧАЮЩЕГОСЯ**  
4 курса группы ПИ-4ИВТ221Б

Санникова Михаила Александровича

Уровень высшего образования:	высшее образование – бакалавриат
Направление подготовки (специальность)	<u>09.03.03 “Прикладная информатика”</u>
Направленность (профиль) программы	<u>Информационные и вычислительные технологии</u>
Дата выполнения	06.10.2025

## **Постановка задачи:**

Создать файл config.ini

Написать функцию для чтения файла

Добавить в файл config.ini:

- Параметр timeout = 30 в секцию settings
- Новую секцию Paths с параметром home = /usr/local/app

Написать функцию, которая проверяет:

- Существование обязательных параметров (host, user в секции Database)
- Корректность типов данных (host — целое число, debug — булево значение)

## Практическая часть:

Листинг кода с комментариями

```
import configparser
import os
from functools import wraps

def main():
    file_path = 'config.ini'

    create_default_config_file(file_path)
    modify_config_file(file_path)

    read_config_file(file_path)

def create_default_config_file(file_path):
    config = configparser.ConfigParser()

    config["database"] = {
        "host": "localhost",
        "port": "5432",
        "user": "admin"
    }
    config["settings"] = {
        "debug": "True",
        "log_level": "INFO"
    }

    with open(file_path, "w", encoding="utf-8") as f:
        config.write(f)

def modify_config_file(file_path):
    config = configparser.ConfigParser()
    config.read(file_path, encoding="utf-8")

    config["settings"]["timeout"] = "30"
    config["Paths"] = {
        "home": "/usr/local/app"
    }

    with open(file_path, "w", encoding="utf-8") as f:
        config.write(f)

def check_file_exists(func):
    @wraps(func)
    def wrapper(file_path):
        if not os.path.exists(file_path):
            print(f'Не существует файла с заданным путем: {file_path}')
            return
        return func(file_path)

    return wrapper
```

```
@check_file_exists
def read_config_file(file_path):
    config = configparser.ConfigParser()
    config.read(file_path, encoding="utf-8")

    if not is_data_valid(config):
        print('Данные некорректны')
        return

    for section in config.sections():
        print(f"\nСекция [{section}]")
        print("-" * 30)
        for key, value in config.items(section):
            print(f"{key} = {value}")
        print("-" * 30)

def is_data_valid(config):
    return (
        config["database"].get("host")
        and config["database"].get("user")
        and config["database"]["port"].isdigit()
        and config["settings"]["debug"].capitalize() in ('True', 'False')
    )

if __name__ == "__main__":
    main()
```

## Скриншоты выполнения программы

Секция [database]

-----

host = localhost

port = 5432

user = admin

-----

Секция [settings]

-----

debug = True

log\_level = INFO

timeout = 30

-----

Секция [Paths]

-----

home = /usr/local/app

-----