**Задание 1: Анализ репозитория и документация по Flaskприложениям**

Цель: Изучить структуру репозитория Flask-apps и описать основные механизмы построения веб-приложений на Flask.   
  
 **Общая структура Flask-приложения**

/flask-app

/static # Статические файлы (CSS, JS, изображения)

/templates # Шаблоны Jinja2

/models # Модели данных (если используется ORM)

app.py # Основной файл приложения

config.py # Конфигурация

requirements.txt # Зависимости

### **Основные механизмы Flask**

#### Маршрутизация (@app.route)

@app.route('/')

def home():

return "Главная страница"

@app.route('/about')

def about():

return "О нас"

#### Обработка запросов (GET, POST)

@app.route('/login', methods=['GET', 'POST'])

def login():

if request.method == 'POST':

# Обработка данных формы

return redirect(url\_for('home'))

return render\_template('login.html')

#### Использование шаблонов (Jinja2)

<!DOCTYPE html>

<html>

<head>

<title>{% block title %}{% endblock %}</title>

</head>

<body>

{% block content %}{% endblock %}

</body>

</html>

Дочерний шаблон:

{% extends "base.html" %}

{% block title %}Главная{% endblock %}

{% block content %}

<h1>Добро пожаловать!</h1>

{% endblock %}

#### Работа с формами (Flask-WTF)

from flask\_wtf import FlaskForm

from wtforms import StringField, TextAreaField

from wtforms.validators import DataRequired, Email

class ContactForm(FlaskForm):

name = StringField('Имя', validators=[DataRequired()])

email = StringField('Email', validators=[DataRequired(), Email()])

message = TextAreaField('Сообщение', validators=[DataRequired()])  
  
Подключение к базе данных (Flask-SQLAlchemy)

from flask\_sqlalchemy import SQLAlchemy

app.config['SQLALCHEMY\_DATABASE\_URI'] = 'sqlite:///site.db'

db = SQLAlchemy(app)

class User(db.Model):

id = db.Column(db.Integer, primary\_key=True)

username = db.Column(db.String(20), unique=True, nullable=False)

email = db.Column(db.String(120), unique=True, nullable=False)

def \_\_repr\_\_(self):

return f"User('{self.username}', '{self.email}')"