|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования**  **«Московский государственный технический университет имени Н.Э. Баумана**  **(национальный исследовательский университет)» (МГТУ им. Н.Э. Баумана)** |

**Факультет «Информатика и системы управления»**

**Кафедра ИУ5 «Системы обработки информации и управления»**

**Отчет по домашнему заданию**

**по дисциплине «Базовые компоненты интернет-технологий»**

Выполнил:

студент группы

ИУ5Ц-52Б Дзауров И.А.

\_\_\_\_\_\_\_\_, “\_\_\_” \_\_\_\_\_\_\_\_\_\_ 2022 г.

Проверил:

преподаватель кафедры

ИУ5 - Гапанюк Ю.Е.

\_\_\_\_\_\_\_\_, “\_\_\_” \_\_\_\_\_\_\_\_\_\_ 2022 г.

Москва, 2022 г.

**Описание задания**

1. Модифицируйте код лабораторной работы №5 или №6 таким образом, чтобы он был пригоден для модульного тестирования.

**Структура проекта:**

main.py

TDDTest.py

**Основной файл - main.py:**

# This is a sample Python script.  
import os  
from aiogram.types import ReplyKeyboardMarkup, KeyboardButton # pip install aiogram  
from aiogram import Dispatcher, Bot, executor, types  
from random import randint, choice  
from string import ascii\_letters, digits, punctuation  
  
API\_TOKEN = "---"  
  
# инициализация бота...  
bot = Bot(token=API\_TOKEN)  
dispatcher = Dispatcher(bot)  
  
  
  
  
# Создание клавиатуры  
btn\_random = KeyboardButton("🎲 Рандомное число")  
btn\_pass = KeyboardButton('🔐 Придумать пароль')  
btn\_other = KeyboardButton("🔷 Другое")  
btn\_photo = KeyboardButton("🔷 Герб")  
  
main\_menu = ReplyKeyboardMarkup(resize\_keyboard=True).add(btn\_random, btn\_pass, btn\_photo, btn\_other)  
  
btn\_info = KeyboardButton('ℹ️ Информация')  
btn\_main = KeyboardButton('🟡 Главное меню')  
  
other\_menu = ReplyKeyboardMarkup(resize\_keyboard=True).add(btn\_info, btn\_main)  
  
@dispatcher.message\_handler(commands=['start'])  
async def start(message: types.Message):  
 await bot.send\_message(message.from\_user.id, f"👋 Привет, {message.from\_user.first\_name}!", reply\_markup=main\_menu)  
  
  
@dispatcher.message\_handler()  
async def messages(message: types.Message):  
 if message.text == '🎲 Рандомное число':  
 await bot.send\_message(message.from\_user.id, f'🎲 Рандомное число: {randint(0, 100)}')  
 elif message.text == '🔐 Придумать пароль':  
 password = "".join([choice(str(digits + ascii\_letters + punctuation)) for \_ in range(24)])  
 await bot.send\_message(message.from\_user.id, f'🔐 Надёжный пароль: {password}')  
 elif message.text == '🔷 Герб':  
 text = 'simple\_text'  
 img\_url = 'Telegram.png'  
 await bot.send\_message(message.from\_user.id, f'{text}<a href="{img\_url}">.</a>', parse\_mode=ParseMode.HTML)  
 elif message.text == '🔷 Другое':  
 await bot.send\_message(message.from\_user.id, '🔷 Открываю...', reply\_markup=other\_menu)  
 elif message.text == 'ℹ️ Информация':  
 await bot.send\_message(message.from\_user.id, f'👽 Ваше имя: {message.from\_user.first\_name}\n\nБот создан by Ibragim 👨‍💻')  
 elif message.text == '🟡 Главное меню':  
 await bot.send\_message(message.from\_user.id, '🟡 Открываю меню...', reply\_markup=main\_menu)  
 else:  
 await bot.send\_message(message.from\_user.id, f'😐 Бот вас не понял... :(')  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 executor.start\_polling(dispatcher, skip\_updates=True)

**TDD-тест - TDDTest.py:**

import unittest  
  
from main import get\_roots, get\_roots\_biquadratic  
  
class TestGetRoots(unittest.TestCase):  
 def testGetRoots(self):  
 self.assertEqual(get\_roots\_biquadratic(get\_roots(4, -5, 1)), [1.0, -1.0, 0.5, -0.5])  
 self.assertEqual(get\_roots\_biquadratic(get\_roots(1, -2, -8)), [2.0, -2.0])  
 self.assertEqual(get\_roots\_biquadratic(get\_roots(1, 1, 1)), [])  
  
 def testValue(self):  
 with self.assertRaises(ValueError) as e:  
 get\_roots\_biquadratic(get\_roots(0, 33, 9))  
  
 def testType(self):  
 with (self.assertRaises(TypeError)) as e:  
 get\_roots\_biquadratic(get\_roots(7, "D", 4))  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 unittest.main()

**Экранные формы с примерами выполнения программы**



