



**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»**

Курс «Базовые компоненты интернет-технологий»  
Отчет по домашнему заданию

«Домашнее задание»

Выполнил:  
студент группы ИУ5-33Б  
Юрова Е.О.

Проверил:  
Канев А.И.

2021 г.

## Задание:

1. Модифицируйте код лабораторной работы №6 таким образом, чтобы он был пригоден для модульного тестирования.
2. Используя материалы лабораторной работы №4 создайте модульные тесты с применением TDD - фреймворка (2 теста) и BDD - фреймворка (2 теста).

## Текст программы:

### Файл main.py

```
# coding=utf-8
import os
import telebot
from telebot import types

# Токент бота
TOKEN = '5013579321:AAG5UtZW97BEzDvSRAoe5a12kHYu0FCrFiQ'

Stray_Kids_songs = ["/Christmas_Evel", "/Domino"]
BTS_songs = ["/Butter", "/Dynamite"]

bot = telebot.TeleBot(TOKEN)
markup = types.ReplyKeyboardMarkup() # табличка кнопочек под полем ввода
сообщения

cur_path = os.path.dirname(os.path.abspath(__file__))
@bot.message_handler(commands=['start'])
def handle_info(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, False)
    bot.send_message(message.chat.id, 'Выберете страну :')
    user_markup.row("/США", "/Южная_Корея")
    bot.send_message(message.chat.id, """"Добро пожаловать :)
    """,
                    reply_markup=user_markup) # ну и отправляем
пользователю сообщение, чтобы он знал, что все работает
# в конце reply_markup = user_markup прикрепляем те самые кнопочки

@bot.message_handler(commands=['Страна'])
def handle_info(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, False)
    user_markup.row("/США", "/Южная_Корея") # первая строка (раз, два)
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
    reply_markup=user_markup)

@bot.message_handler(commands=['США'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, False)
    bot.send_message(message.chat.id, 'Выберете исполнителя:')
    user_markup.row("/Imagine_Dragons", "/One_Direction")
    user_markup.row("/Страна")
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
    reply_markup=user_markup)

@bot.message_handler(commands=['Южная_Корея'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, False)
    bot.send_message(message.chat.id, 'Выберете исполнителя:')
    user_markup.row("/BTS", "/Stray_Kids")
```

```

        user_markup.row("/Страна")
        bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
reply_markup=user_markup)

@bot.message_handler(commands=['BTS'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, True)
    bot.send_message(message.chat.id, 'Выберете песню:')
    user_markup.row("/Butter", "/Dynamite")
    user_markup.row("/Южная Корея")
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
reply_markup=user_markup)

@bot.message_handler(commands=['Stray_Kids'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, True)
    bot.send_message(message.chat.id, 'Выберете песню:')
    user_markup.row("/Christmas_Evel", "/Domino")
    user_markup.row("/Южная Корея")
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
reply_markup=user_markup)

@bot.message_handler(commands=['Imagine_Dragons'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, True)
    bot.send_message(message.chat.id, 'Выберете песню:')
    user_markup.row("/Believer", "/Thunder")
    user_markup.row("/США")
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
reply_markup=user_markup)

@bot.message_handler(commands=['One_Direction'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, True)
    bot.send_message(message.chat.id, 'Выберете песню:')
    user_markup.row("/Illusion", "/Wolves")
    user_markup.row("/США")
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
reply_markup=user_markup)

@bot.message_handler(commands=['Butter'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, True)
    bot.send_audio(message.chat.id, open(
"/Users/evgeniayurova/Desktop/video/bts-butter.mp3",'rb'))
    user_markup.row("/BTS")
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку ",
reply_markup=user_markup)

@bot.message_handler(commands=['Dynamite'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, True)
    bot.send_audio(message.chat.id, open(
"/Users/evgeniayurova/Desktop/video/BTS-Dynamite_muzrecord.net.mp3",'rb'))
    user_markup.row("/BTS")
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
reply_markup=user_markup)

@bot.message_handler(commands=['Wolves'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, True)
    bot.send_audio(message.chat.id, open(
"/Users/evgeniayurova/Desktop/video/One Direction - Wolves
[vzvuke.net].mp3",'rb'))

```

```

        user_markup.row("/One_Direction")
        bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
reply_markup=user_markup)

@bot.message_handler(commands=['Illusion'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, True)
    bot.send_audio(message.chat.id,
                    open("/Users/evgeniayurova/Desktop/video/One Direction -
Illusion [vzvuke.net].mp3", 'rb'))
    user_markup.row("/One_Direction")
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
reply_markup=user_markup)

@bot.message_handler(commands=['Thunder'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, True)
    bot.send_audio(message.chat.id,
                    open("/Users/evgeniayurova/Desktop/video/Imagine Dragons -
Thunder.mp3", 'rb'))
    user_markup.row("/Imagine_Dragons")
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
reply_markup=user_markup)

@bot.message_handler(commands=['Believer'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, True)
    bot.send_audio(message.chat.id,
                    open("/Users/evgeniayurova/Desktop/video/Imagine Dragons -
Believer.mp3", 'rb'))
    user_markup.row("/Imagine_Dragons")
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
reply_markup=user_markup)

@bot.message_handler(commands=['Domino'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, True)
    bot.send_audio(message.chat.id,
                    open("/Users/evgeniayurova/Desktop/video/Stray Kids -
Domino.mp3", 'rb'))
    user_markup.row("/Stray_Kids")
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
reply_markup=user_markup)

@bot.message_handler(commands=['Christmas_Evel'])
def handle_stop(message):
    user_markup = telebot.types.ReplyKeyboardMarkup(True, True)

    bot.send_audio(message.chat.id, open("/Users/evgeniayurova/Desktop/video/stray
-kids-christmas-evil-mp3.mp3", 'rb'))
    user_markup.row("/Stray_Kids")
    bot.send_message(message.chat.id, "Пожалуйста нажмите кнопку",
reply_markup=user_markup)

def check_bts(text):
    if text in BTS_songs:
        return 1
    else:
        return 0

def check_stray(text):
    if text in Stray_Kids_songs:
        return 1

```

```

        else:
            return 0

bot.infinity_polling()

```

### Файл tdd.py

```

import unittest
import sys, os

sys.path.append(os.getcwd())
from main import *

class myTest(unittest.TestCase):
    def test_1(self):
        pass
        self.assertEqual(check_bts("/Butter"), 1)
        self.assertEqual(check_bts("/Wolves"), 0)
    def test_2(self):
        pass
        self.assertEqual(check_stray("/Domino"), 1)
        self.assertEqual(check_stray("/Dynamite"), 0)

```

### Файл feature.feature

```

Feature: Test
  Scenario: Bot testing
    Given Bot
    When test_1 return OK
    And test_2 return OK
    Then all right

```

### Файл bdd.py

```

from behave import *

from tdd import *

@given('Bot')
def first_step(context):
    context.a = myTest()

@when('test_1 return OK')
def test_1(context):
    context.a.test_1()

@when('test_2 return OK')
def test_2(context):
    context.a.test_2()

@then('all right')
def last_step(context):
    pass

```

**Экранные формы с примерами выполнения программы:**

Результаты работы tdd.py:

```
Tests passed: 2 of 2 tests - 1 ms
1 ms
Testing started at 21:39 ...
Launching unittests with arguments python -m unittest tdd.myTest in /Users/evgeniayurova/PycharmProjects/Lab6

2021-12-17 21:39:09,728 (__init__.py:629 MainThread) ERROR - TeleBot: "Break infinity polling"

Ran 2 tests in 0.002s

OK

Process finished with exit code 0
```

blems Terminal Python Packages Python Console