



## Отчет по домашней работе

Студент	ИУ5Ц-52Б (Группа)		А.Н. Свинцов (И.О. Фамилия)
Преподаватель			Ю.Е. Гапанюк (И.О. Фамилия)

Москва, 2021

## Описание задания

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).

## Код программы

bot2.py:

```
from aiogram import Bot, types
from aiogram.dispatcher import Dispatcher
from aiogram.dispatcher.filters.state import StatesGroup, State
from aiogram.dispatcher.filters import Command
from aiogram.dispatcher.storage import FSMContext
from aiogram.utils import executor
from config import TOKEN
from aiogram.types import ReplyKeyboardRemove, ReplyKeyboardMarkup,
KeyboardButton, InlineKeyboardMarkup, InlineKeyboardButton, Message

from aiogram.contrib.fsm_storage.memory import MemoryStorage
from aiogram.contrib.middlewares.logging import LoggingMiddleware

bot = Bot(token=TOKEN)
dp = Dispatcher(bot, storage=MemoryStorage())
dp.middleware.setup(LoggingMiddleware())

class Test(StatesGroup):
    Q0 = State()
    Q1 = State()
    Q2 = State()
    Q3 = State()

material_price={
    "Дерево" : 65,
    "Кафель" : 25,
    "Ламинат" : 70
}

equip_price={
    "Молоток" : 250,
    "Дрель" : 570,
    "Лобзик" : 450
}

fasteners_price={
    "Винт" : 35,
    "Болт" : 20,
    "Гайка" : 35
}

material={
    1 : "Дерево",
    2 : "Кафель",
    3 : "Ламинат"
}

equip={
    1 : "Молоток",
    2 : "Дрель",
```

```

        3 : "Лобзик"
    }
    fasteners={
        1 : "Винт",
        2 : "Болт",
        3 : "Гайка"
    }

def
summary(first_table,first_table_price,second_table,second_table_price,third_t
able,third_table_price,first_answer,second_answer,third_answer):
    a = first_table_price[first_table[first_answer]]
    b = second_table_price[second_table[second_answer]]
    c = third_table_price[third_table[third_answer]]
    return "Итоговая сумма = " + str(a+b+c)+ "\n"

@dp.message_handler(state="*", commands=['start'])
async def starting_process(message: types.Message):
    await bot.send_message(message.from_user.id,"Приветствуем вас в нашем
интернет магазине.\nЗдесь вы можете заказать строй товары из следующих
категорий:\n \
    1)Материалы\n \
    2)Оборудование\n \
    3)Крепежи\nДля формирования заказа /order")
    await Test.Q0.set()

@dp.message_handler(state=Test.Q0, commands=['order'])
async def starting_process(message: types.Message,state: FSMContext):
    await bot.send_message(message.from_user.id, "Выберите материал\n1) Дерево
- 50\n2) Кафель - 75\n3) Ламинат - 100")
    await Test.Q1.set()

@dp.message_handler(state=Test.Q1)
async def first_choosing(message: types.Message,state: FSMContext):
    answer = int(message.text)
    if (answer != 1 and answer !=2 and answer !=3):
        return await bot.send_message(message.from_user.id,"К сожалению,
такого товара нет в наличии, попробуйте выбрать другой")
    await state.update_data(q1 = answer)
    await bot.send_message(message.from_user.id,"Выберете
оборудование\n1) Молоток - 250\n2) Дрель - 500\n3) Лобзик - 350")
    await Test.Q2.set()

@dp.message_handler(state=Test.Q2)
async def second_choosing(message: types.Message,state: FSMContext):
    answer = int(message.text)
    if (answer != 1 and answer !=2 and answer !=3):
        return await bot.send_message(message.from_user.id,"К сожалению,
такого товара нет в наличии, попробуйте выбрать другой")
    await state.update_data(q2 = answer)
    await bot.send_message(message.from_user.id, "Выберите крепеж\n1) Винт -
35 \n2) Болт - 20 \n3) Гайка - 10")
    await Test.Q3.set()

@dp.message_handler(state=Test.Q3)
async def third_choosing(message: types.Message,state: FSMContext):
    answer = int(message.text)
    if (answer != 1 and answer !=2 and answer !=3):
        return await bot.send_message(message.from_user.id,"К сожалению,

```

```

такого товара нет в наличии, попробуйте выбрать другой")
    await state.update_data(q3 = answer)
    data = await state.get_data()

sumcheck=summary(material,material_price,equip,equip_price,fasteners,fastener
s_price,data.get("q1"),data.get("q2"),data.get("q3"))
    await
bot.send_message(message.from_user.id,"Итого:\nМатериал:\n{}\nОборудование:\n
{}\nКрепёж:\n{}".format(material[data.get("q1")],equip[data.get("q2")],fasten
ers[data.get("q3")]))
    await bot.send_message(message.from_user.id, sumcheck)
    await Test.Q0.set()

async def shutdown(dispatcher: Dispatcher):
    await dispatcher.storage.close()
    await dispatcher.storage.wait_closed()

if __name__ == '__main__':
    executor.start_polling(dp, on_shutdown=shutdown)

```

## confit.py:

```
TOKEN="*****"
```

## test1.py:

```

import unittest
import sys, os

sys.path.append(os.getcwd())
from bot2 import *

test1 = 1
test2 = 2
test3 = 3

class TestGetRoots(unittest.TestCase):
    def test1_bot(self):
        res =
summary(material,material_price,equip,equip_price,fasteners,fasteners_price,t
est1,test2,test3)
        self.assertEqual("Итого = 670\n", res)
    def test2_bot(self):
        res =
summary(material,material_price,equip,equip_price,fasteners,fasteners_price,t
est2,test1,test3)
        self.assertEqual("Итого = 310\n", res)

if __name__ == "__main__":
    unittest.main()

```

## Экранные формы

```
C:\Users\User\PycharmProjects\02\venv\Scripts\python.exe C:/Users/User/PycharmProjects/02/test1.p
..
-----
Ran 2 tests in 0.000s

OK

Process finished with exit code 0
```