

**Московский государственный технический университет
им. Н.Э. Баумана**

Факультет ИУ Кафедра ИУ5

**Курс «Основы информатики» Отчет
по Рубежному контролю №2**

Выполнил студент группы ИУ5-33Б:

Ветлугин И.Г.

Подпись и дата:

Проверил преподаватель каф.:

Гапанюк Ю. Е.

Подпись и дата:

Москва, 2024 г

Постановка задачи

Рубежный контроль представляет собой разработку тестов на языке Python.

- 1) Проведите рефакторинг текста программы рубежного контроля №1 таким образом, чтобы он был пригоден для модульного тестирования.
- 2) Для текста программы рубежного контроля №1 создайте модульные тесты с применением TDD - фреймворка (3 теста).

Текст программы

rk1_refactored.py

```
from collections import Counter

class Column:
    def __init__(self, name, data_type):
        self.name = name
        self.data_type = data_type

class Table:
    def __init__(self, name, columns):
        self.name = name
        self.columns = columns

employee_columns = [
    Column('id', 'int'),
    Column('last_name', 'varchar'),
    Column('salary', 'float'),
    Column('department_id', 'int')
]

department_columns = [
    Column('id', 'int'),
    Column('name', 'varchar')
]

employees_table = Table('Employee', employee_columns)
departments_table = Table('Department', department_columns)

employee_department_columns = [
    Column('employee_id', 'int'),
    Column('department_id', 'int')
]

employee_departments_table = Table('Employee_Department', employee_department_columns)

def get_employees_and_departments(employees, departments):
    result = sorted(
        [(emp['last_name'], dept['name']) for emp in employees for dept in departments
         if emp['department_id'] == dept['id']],
        key=lambda x: x[0]
    )
    return result

def get_departments_with_employee_count(employees, departments):
    employee_counts = Counter([dept['id'] for emp in employees for dept in departments
```

```

if emp['department_id'] == dept['id']]
    result = sorted(
        [(dept['name'], employee_counts.get(dept['id'], 0)) for dept in departments],
        key=lambda x: x[1], reverse=True
    )
    return result

def get_employees_last_name_ov(employees, departments):
    ov_employees = [emp for emp in employees if emp['last_name'].endswith('ов')]
    result = [(emp['last_name'], dept['name'])
               for emp in ov_employees
               for dept in departments
               if emp['department_id'] == dept['id']]
    return result

```

module_tests.py

```

import pytest
from rkl_refactored import (
    get_employees_and_departments,
    get_departments_with_employee_count,
    get_employees_last_name_ov
)

@pytest.fixture
def employees():
    return [
        {'id': 1, 'last_name': 'Иванов', 'salary': 50000, 'department_id': 1},
        {'id': 2, 'last_name': 'Петров', 'salary': 60000, 'department_id': 2},
        {'id': 3, 'last_name': 'Сидоров', 'salary': 70000, 'department_id': 1}
    ]

@pytest.fixture
def departments():
    return [
        {'id': 1, 'name': 'Продажи'},
        {'id': 2, 'name': 'Маркетинг'}
    ]

def test_get_employees_and_departments(employees, departments):
    expected_result = [
        ('Иванов', 'Продажи'),
        ('Петров', 'Маркетинг'),
        ('Сидоров', 'Продажи')
    ]
    assert get_employees_and_departments(employees, departments) == expected_result

def test_get_departments_with_employee_count(employees, departments):
    expected_result = [
        ('Продажи', 2),
        ('Маркетинг', 1)
    ]
    assert get_departments_with_employee_count(employees, departments) == expected_result

def test_get_employees_last_name_ov(employees, departments):
    expected_result = [
        ('Иванов', 'Продажи'),
        ('Петров', 'Маркетинг'),
        ('Сидоров', 'Продажи')
    ]
    assert get_employees_last_name_ov(employees, departments) == expected_result

```

Анализ результатов

```
Testing started at 23:05 ...
Launching pytest with arguments D:\etc\docs\PCPL\RK2\module_tests.py --no-header --no-summary -q in D:\etc\docs\PCPL\RK2

===== test session starts =====
collecting ... collected 3 items

module_tests.py::test_get_employees_and_departments PASSED          [ 33%]
module_tests.py::test_get_departments_with_employee_count PASSED    [ 66%]
module_tests.py::test_get_employees_last_name_ov PASSED            [100%]

===== 3 passed in 0.01s =====

Process finished with exit code 0
```