

Московский государственный технический университет
им. Н.Э. Баумана

Факультет “Информатика и системы управления”
Кафедра “Системы обработки информации и управления”



Дисциплина «Парадигмы и конструкции языков программирования»

Отчет по рубежному контролю №1

Выполнил:
студент группы ИУ5-36Б
Милов Е.Д.

Проверил:
Нардид Анатолий Николаевич

Москва, 2025

Условия рубежного контроля №1 по курсу ПиК ЯП

Рубежный контроль представляет собой разработку программы на языке Python, которая выполняет следующие действия:

1) Необходимо создать два класса данных в соответствии с Вашим вариантом предметной области, которые связаны отношениями один-ко-многим и многие-ко-многим.

Пример классов данных для предметной области Сотрудник-Отдел:

1. Класс «Сотрудник», содержащий поля:

- ID записи о сотруднике;
- Фамилия сотрудника;
- Зарплата (количественный признак);
- ID записи об отделе. (для реализации связи один-ко-многим)

2. Класс «Отдел», содержащий поля:

- ID записи об отделе;
- Наименование отдела.

3. (Для реализации связи многие-ко-многим) Класс «Сотрудники отдела», содержащий поля:

- ID записи о сотруднике;
- ID записи об отделе.

2) Необходимо создать списки объектов классов, содержащих тестовые данные (3-5 записей), таким образом, чтобы первичные и вторичные ключи соответствующих записей были связаны по идентификаторам.

3) Необходимо разработать запросы в соответствии с Вашим вариантом. Запросы сформулированы в терминах классов «Сотрудник» и «Отдел», которые используются в примере. Вам нужно перенести эти требования в Ваш вариант предметной области. При разработке запросов необходимо по возможности использовать функциональные возможности языка Python (list/dict comprehensions, функции высших порядков). Для реализации запроса №2 введите в класс, находящийся на стороне связи «много», произвольный количественный признак, например, «зарплата сотрудника». Результатом рубежного контроля является документ в формате PDF, который содержит текст программы и результаты ее выполнения.

Мой вариант запросов - Вариант Д.

1. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим.
Выведите список всех сотрудников, у которых фамилия
заканчивается на «ов», и названия их отделов
2. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим.
Выведите список отделов с со средней зарплатой сотрудников в
каждом отделе, отсортированный по средней зарплате.
3. «Отдел» и «Сотрудник» связаны соотношением многие-ко-
многим. Выведите список всех отделов, у которых название
начинается с буквы «А», и список работающих в них
сотрудников.

**Мои варианты предметной области - Вариант 12 (Язык
программирования, средство разработки)**

Текст программы

```
class ProgrammingLanguage:
    def __init__(self, id, name):
        self.id = id
        self.name = name

class DevelopmentTool:
    def __init__(self, id, name, license_cost, lang_id=None):
        self.id = id
        self.name = name
        self.license_cost = license_cost
        self.lang_id = lang_id

class ToolLanguage:
    def __init__(self, tool_id, lang_id):
        self.tool_id = tool_id
        self.lang_id = lang_id

langs = [
    ProgrammingLanguage(1, "Python"),
    ProgrammingLanguage(2, "Java"),
    ProgrammingLanguage(3, "C++"),
    ProgrammingLanguage(4, "JavaScript"),
    ProgrammingLanguage(5, "Ada"), # Добавим язык на "A"
]

tools = [
    DevelopmentTool(1, "PyCharm Professional", 199, 1),
    DevelopmentTool(2, "IntelliJ IDEA Ultimate", 249, 2),
    DevelopmentTool(3, "Visual Studio Enterprise", 299, 3),
    DevelopmentTool(4, "VS Code", 0, 4),
    DevelopmentTool(5, "Eclipse IDE", 0, 2),
    DevelopmentTool(6, "GNAT Programming Studio", 150, 5), # Для языка Ada
    DevelopmentTool(7, "AdaCore", 199, 5),
]

tool_langs = [
    ToolLanguage(1, 1),
    ToolLanguage(2, 2),
    ToolLanguage(3, 3),
    ToolLanguage(4, 4),
    ToolLanguage(4, 1),
    ToolLanguage(4, 2),
    ToolLanguage(5, 2),
    ToolLanguage(5, 3),
    ToolLanguage(6, 5),
    ToolLanguage(7, 5),
]

def main():
    print("Запрос 1:")
    query1 = [(tool.name, next(lang.name for lang in langs if lang.id == tool.lang_id))
               for tool in tools if tool.name.endswith('ov') or tool.name.endswith('OV')]
    print("Средства разработки с названиями на 'ов':", query1)

    print("\nЗапрос 1 (альтернатива - окончание 'Studio'):")
    query1_alt = [(tool.name, next(lang.name for lang in langs if lang.id == tool.lang_id))
                  for tool in tools if tool.name.endswith('Studio')]
    print(query1_alt)
```

```
print("\nЗапрос 2:")
lang_tool_stats = {}
for lang in langs:
    lang_tools = [tool for tool in tools if tool.lang_id == lang.id]
    if lang_tools:
        total_cost = sum(tool.license_cost for tool in lang_tools)
        count = len(lang_tools)
        avg_cost = total_cost / count
        lang_tool_stats[lang.name] = avg_cost

query2 = sorted(lang_tool_stats.items(), key=lambda x: x[1])
print("Языки программирования со средней стоимостью инструментов:")
for lang, avg_cost in query2:
    print(f"{lang}: {avg_cost:.2f}")

print("\nЗапрос 3:")
query3 = []
for lang in langs:
    if lang.name.startswith('A'):
        supporting_tools = []
        for tl in tool_langs:
            if tl.lang_id == lang.id:
                tool = next((t for t in tools if t.id == tl.tool_id), None)
                if tool:
                    supporting_tools.append(tool.name)
        query3.append((lang.name, supporting_tools))

print("Языки на 'А' и их инструменты (многие-ко-многим):")
for lang_name, tool_list in query3:
    print(f"{lang_name}: {' '.join(tool_list)}")

if __name__ == "__main__":
    main()
```

Результаты выполнения программы

```
"C:\Users\Ирина Милова\PycharmProjects\Python_Labs\.venv\Scripts\python.exe" "C:\Users\Ирина Милова\PycharmProjects\Python_Labs\RК1\main.py"
Запрос 1:
Средства разработки с названиями на 'ов': []

Запрос 1 (альтернатива - окончание 'Studio'):
[('GNAT Programming Studio', 'Ada')]

Запрос 2:
Языки программирования со средней стоимостью инструментов:
JavaScript: 0.00
Java: 124.50
Ada: 174.50
Python: 199.00
C++: 299.00

Запрос 3:
Языки на 'А' и их инструменты (многие-ко-многим):
Ada: GNAT Programming Studio, AdaCore

Process finished with exit code 0
```

