

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
Степанов А. 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 15** (Файл, каталог файлов)

Текст программы

```
from dataclasses import dataclass
from typing import List
from operator import itemgetter

@dataclass
class FileCatalog:
    id: int
    name: str

@dataclass
class File:
    id: int
    name: str
    size: int
    catalog_id: int

@dataclass
class FileCatalogFile:
    catalog_id: int
    file_id: int

catalogs = [
    FileCatalog(1, "Академические документы"),
    FileCatalog(2, "Архив проектов"),
    FileCatalog(3, "Городская документация"),
    FileCatalog(4, "Научные работы")
]

files = [
    File(1, "отчет_иванов.doc", 5, 1),
    File(2, "презентация_петров.ppt", 12, 2),
    File(3, "статья_сидоров.pdf", 3, 1),
    File(4, "документ_смит.docx", 7, 3),
    File(5, "проект_джонс.zip", 25, 2),
    File(6, "исследование.bin", 15, 4),
    File(7, "анализ_данных.rar", 18, 4),
    File(8, "финальный_отчет.pdf", 8, 1),
    File(9, "бюджетов.xlsx", 6, 3),
    File(10, "результатов.csv", 4, 4),
    File(11, "отчетов.doc", 9, 1),
]

catalog_files = [
    FileCatalogFile(1, 1),
    FileCatalogFile(1, 3),
    FileCatalogFile(2, 2),
    FileCatalogFile(2, 5),
    FileCatalogFile(3, 4),
    FileCatalogFile(4, 6),
    FileCatalogFile(4, 7),
    FileCatalogFile(1, 8),
```

```

FileCatalogFile(2, 4),
FileCatalogFile(3, 9),
FileCatalogFile(4, 10),
FileCatalogFile(1, 11),
]

def main():
    print("Каталоги файлов:")
    for catalog in catalogs:
        print(f" {catalog.id}. {catalog.name}")

    print("\nФайлы:")
    for file in files:
        print(f" {file.id}. '{file.name}' - {file.size} МБ")

    print("\nСвязи файлов с каталогами:")
    for cf in catalog_files:
        catalog_name = next((catalog.name for catalog in catalogs if catalog.id ==
cf.catalog_id), "Неизвестно")
        file_name = next((file.name for file in files if file.id == cf.file_id),
"Неизвестно")
        print(f" Каталог '{catalog_name}' -> Файл '{file_name}'")

    one_to_many = [(file.name, file.size, catalog.name)
                    for catalog in catalogs
                    for file in files
                    if file.catalog_id == catalog.id]

    many_to_many_temp = [(catalog.name, cf.catalog_id, cf.file_id)
                          for catalog in catalogs
                          for cf in catalog_files
                          if catalog.id == cf.catalog_id]

    many_to_many = [(file.name, file.size, catalog_name)
                    for catalog_name, catalog_id, file_id in many_to_many_temp
                    for file in files if file.id == file_id]

    print("\nЗАПРОС 1: Файлы, названия которых заканчиваются на 'ов', и их
каталоги")

    files_ending_with_ov = []
    for file in files:
        file_name_without_extension = file.name.rsplit('.', 1)[0]
        if file_name_without_extension.endswith('ов'):
            catalog_name = next((catalog.name for catalog in catalogs if catalog.id
== file.catalog_id), "Неизвестный каталог")
            files_ending_with_ov.append((file.name, file.size, catalog_name))

    if files_ending_with_ov:
        for file_name, file_size, catalog_name in files_ending_with_ov:
            print(f" '{file_name}' - {file_size} МБ")
            print(f" Находится в: {catalog_name}")
    else:

```

```

    print("  Файлов с названиями, оканчивающимися на 'ов', не найдено")

print("\nЗАПРОС 2: Каталоги со средним размером файлов")

catalog_avg_sizes = []
for catalog in catalogs:
    catalog_files_list = [file for file in files if file.catalog_id ==
catalog.id]
    if catalog_files_list:
        total_size = sum(file.size for file in catalog_files_list)
        avg_size = total_size / len(catalog_files_list)
        catalog_avg_sizes.append((catalog.name, avg_size,
len(catalog_files_list)))

catalog_avg_sizes_sorted = sorted(catalog_avg_sizes, key=itemgetter(1))

for catalog_name, avg_size, file_count in catalog_avg_sizes_sorted:
    print(f"  {catalog_name}:")
    print(f"    Средний размер файлов: {avg_size:.1f} МБ")
    print(f"    Количество файлов: {file_count}")

print("\nЗАПРОС 3: Каталоги с названием на 'А' и их файлы")

catalogs_starting_with_a = [catalog for catalog in catalogs if
catalog.name.startswith('А')]

for catalog in catalogs_starting_with_a:
    print(f"\n  Каталог: {catalog.name}")

    catalog_file_ids = [cf.file_id for cf in catalog_files if cf.catalog_id ==
catalog.id]
    catalog_files_list = [file for file in files if file.id in
catalog_file_ids]

    if catalog_files_list:
        for file in catalog_files_list:
            print(f"    - '{file.name}' - {file.size} МБ")
    else:
        print("    В этом каталоге пока нет файлов")

if __name__ == '__main__':
    main()

```

Результаты выполнения программы

```
winer@NoytPooha:~/sem3/python/rk1$ python3 rk1.py
```

```
Каталог 'Городская документация' -> Файл 'бюджетов.xlsx'  
Каталог 'Научные работы' -> Файл 'результатов.csv'  
Каталог 'Академические документы' -> Файл 'отчетов.doc'
```

ЗАПРОС 1: Файлы, названия которых заканчиваются на 'ов', и их каталоги

```
'отчет_иванов.doc' - 5 МБ  
  Находится в: Академические документы  
'презентация_петров.ppt' - 12 МБ  
  Находится в: Архив проектов  
'статья_сидоров.pdf' - 3 МБ  
  Находится в: Академические документы  
'бюджетов.xlsx' - 6 МБ  
  Находится в: Городская документация  
'результатов.csv' - 4 МБ  
  Находится в: Научные работы  
'отчетов.doc' - 9 МБ  
  Находится в: Академические документы
```

ЗАПРОС 2: Каталоги со средним размером файлов

```
Академические документы:  
  Средний размер файлов: 6.2 МБ  
  Количество файлов: 4  
Городская документация:  
  Средний размер файлов: 6.5 МБ  
  Количество файлов: 2  
Научные работы:  
  Средний размер файлов: 12.3 МБ  
  Количество файлов: 3  
Архив проектов:  
  Средний размер файлов: 18.5 МБ  
  Количество файлов: 2
```

ЗАПРОС 3: Каталоги с названием на 'А' и их файлы

```
Каталог: Академические документы  
  - 'отчет_иванов.doc' - 5 МБ  
  - 'статья_сидоров.pdf' - 3 МБ  
  - 'финальный_отчет.pdf' - 8 МБ  
  - 'отчетов.doc' - 9 МБ
```

```
Каталог: Архив проектов  
  - 'презентация_петров.ppt' - 12 МБ  
  - 'документ_смит.docx' - 7 МБ  
  - 'проект_джонс.zip' - 25 МБ
```

```
winer@NoytPooha:~/sem3/python/rk1$ python3 rk1.py
```