

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 ИУ5-34Б
Шерстеникин Н.С.
Преподаватель:
Гапанюк Ю.Е.

Москва 2025

```

class Stroki:
    def __init__(self, id, content, CharAmount, TableId):
        self.id = id
        self.content = content
        self.CharAmount = CharAmount
        self.TableId = TableId

    def __repr__(self):
        return f"Stroki(id={self.id}, content='{self.content}',
chars={self.CharAmount}, table_id={self.TableId})"

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

    def __repr__(self):
        return f"Table(id={self.id}, name='{self.name}')"

class TableString:
    def __init__(self, id, StringId, TableId):
        self.id = id
        self.StringId = StringId
        self.TableId = TableId

    def __repr__(self):
        return f"TableString(id={self.id}, string_id={self.StringId},
table_id={self.TableId})"

StrokiList = [
    Stroki(1, "бла-бла", 7, 1),
    Stroki(2, "барбер", 6, 2),
    Stroki(3, "ода", 3, 3),
    Stroki(4, "гильгамеш", 9, 3)
]

tables = [
    Table(1, "текст"),
    Table(2, "мало текста"),
    Table(3, "много текста")
]

TableStrings = [
    TableString(1, 1, 1),
    TableString(2, 2, 2),

```

```

    TableString(3, 3, 3),
    TableString(4, 4, 3),
    TableString(5,3,2) #Добавил эту строку сюда,чтобы показать связь
многие-ко-многим
]

```

```

print("1. Строки, начинающиеся с 'б' и их таблицы:")
result1 = []
for stroka in StrokiList:
    if stroka.content.startswith('б'):
        table = next((t for t in tables if t.id == stroka.TableId), None)
        if table:
            result1.append((stroka.content, table.name))

```

```

for content, TableName in result1:
    print(f"  Строка: '{content}', Таблица: '{TableName}'")

```

```

print("\n2. Минимальное количество символов в строках таблиц
(сортировка по возрастанию):")

```

```

MinCharsInTable = { }
for table in tables:
    TableStrokiList = [s for s in StrokiList if s.TableId == table.id]
    if TableStrokiList:
        MinChars = min(s.CharAmount for s in TableStrokiList)
        MinCharsInTable[table.name] = MinChars

```

```

SortedMinChars = sorted(MinCharsInTable.items(), key=lambda x: x[1])
for TableName, MinChars in SortedMinChars:
    print(f"  Таблица '{TableName}': {MinChars} символов")

```

```

print("\n3. Все строки и их таблицы (сортировка по ID строки):")
result3 = []
for ts in TableStrings:
    stroka = next((s for s in StrokiList if s.id == ts.StringId), None)
    table = next((t for t in tables if t.id == ts.TableId), None)
    if stroka and table:
        result3.append((stroka.id, stroka.content, table.name))

```

```

result3_sorted = sorted(result3, key=lambda x: x[0])
for StringId, content, TableName in result3_sorted:
    print(f"  ID строки: {StringId}, Содержание: '{content}', Таблица:
'{TableName}'")

```

Работа кода:

1. Строки, начинающиеся с 'б' и их таблицы:
Строка: 'бла-бла', Таблица: 'текст'
Строка: 'барбер', Таблица: 'мало текста'
2. Минимальное количество символов в строках таблиц (сортировка по возрастанию):
Таблица 'много текста': 3 символов
Таблица 'мало текста': 6 символов
Таблица 'текст': 7 символов
3. Все строки и их таблицы (сортировка по ID строки):
ID строки: 1, Содержание: 'бла-бла', Таблица: 'текст'
ID строки: 2, Содержание: 'барбер', Таблица: 'мало текста'
ID строки: 3, Содержание: 'ода', Таблица: 'много текста'
ID строки: 3, Содержание: 'ода', Таблица: 'мало текста'
ID строки: 4, Содержание: 'гильгамеш', Таблица: 'много текста'