ИУ5-31Б Оразов А.В.
Отчет по РК1

Вариант 14А

Текст программы:

```
class CDDrive:
    def __init__(self, id, content, memory_used_mb, lib_id):
        self.id = id
        self.content = content
        self.memory_used_mb = memory_used_mb
        self.lib_id = lib_id


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


class CDLibraryDrives:
    def __init__(self, drive_id, lib_id):
        self.drive_id = drive_id
        self.lib_id = lib_id


libs = [
    CDLibrary(1, 'music'),
    CDLibrary(2, 'films'),
    CDLibrary(3, 'games'),
    CDLibrary(4, 'books'),
    CDLibrary(42, 'blank')
]

drives = [
    CDDrive(1, 'Queen "Made In Heaven" album', 500, 1),
    CDDrive(2, 'Heat (1995)', 680, 2),
    CDDrive(3, 'Doom Eternal (cd 1 of 1171)', 700, 3),
    CDDrive(4, 'Doom Eternal (cd 2 of 1171)', 700, 3),
    CDDrive(1174, 'George R. R. Martin "A Song of Ice and Fire" (1-5)', 50, 4),
    CDDrive(1175, 'Jack London "Love of Life" (storybook)', 10, 4),
    CDDrive(1176, 'Clean drive #2773', 0, 42)
]

drives_libs = [
    CDLibraryDrives(1, 1),
    CDLibraryDrives(2, 2),
    CDLibraryDrives(3, 3),
    CDLibraryDrives(4, 3),
    CDLibraryDrives(1174, 4),
    CDLibraryDrives(1175, 4),
```

```
    CDLibraryDrives(1176, 42),

    CDLibraryDrives(3, 1),
    CDLibraryDrives(4, 1)
]
```

```
def main():
    # one-to-many list comprehension
    one_to_many = [(d.content, d.memory_used_mb, l.name)
                   for l in libs
                   for d in drives
                   if d.lib_id == l.id]

    print('Задание A1')
    res_11 = sorted(one_to_many, key=lambda tuple: tuple[2])
    print(res_11)

    print('\nЗадание A2')
    # one-to-many list aggregation by "memory_used_kb" field
    res_12_unsorted = [(l.name, sum(otm[1]
                                    for otm in one_to_many
                                    if otm[2] == l.name
                                    ))
                       for l in libs]
    res_12 = sorted(res_12_unsorted, key=lambda l: l[0], reverse=True)
    print(res_12)

    print('\nЗадание A3')
    # many-to-many nested list comprehension
    res_13 = [(l.name, [d.content
                        for d in drives
                        for dl in drives_libs
                        if dl.lib_id == l.id and dl.drive_id == d.id
                        ])
              for l in libs
              if 'b' in l.name]
    print(res_13)


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

Вывод:

```
[('Clean drive #2773', 0, 'blank'), ('George R. R. Martin "A Song of Ice and Fire" (1-5)', 50, 'books'),
('Jack London "Love of Life" (storybook)', 10, 'books'), ('Star wars IV', 680, 'films'), ('Doom Eternal
(cd 1 of 1171)', 700, 'games'), ('Doom Eternal (cd 2 of 1171)', 700, 'games'), ('Queen "Made In Heaven"
album', 500, 'music')]

Задание A2
[('music', 500), ('games', 1400), ('films', 680), ('books', 60), ('blank', 0)]

Задание A3
[('books', ['George R. R. Martin "A Song of Ice and Fire" (1-5)', 'Jack London "Love of
Life" (storybook)']), ('blank', ['Clean drive #2773'])]
```