

Lab 1 Part 2 Writeup

Team:

Evin Killian, Tyler Campanile

Design Decisions:

We chose to reuse our code from part 1 and implement the new features into this code. We created a Teacher class to represent each teacher (first name, last name, and the classroom they teach in). We store all the Teacher objects in a list, like how we do for Student objects. As such, the only parts affected were the file IO/data initialization, and the options to choose from.

For NR1 and NR2, we added the following option:

C[lassroom] <number>

This option outputs all the students in the specified classroom and the teachers who teach in it.

For NR3, we modified the Grade option to output the list of teachers who teach in the specified grade after outputting the students.

For NR4, we added the following option:

E[nrollment]

Which lists the classroom number followed by the number of students enrolled in that classroom.

For the analysis tools, we created a new set of options that allow the user to specify whether they want to analyze grade level, teacher, or bus route.

The new analysis tools give the following options:

An[alysis] <G[rade] | T[eacher] | B[us]>

Each option lists the averages for that query (i.e. Grade will list the average GPA per grade, Teacher will list the average GPA per teacher, and Bus will list the average GPA per bus)

Task Log:

Task Name	Started by	Start Time	End Time	Total Hours
NR1	Tyler	4/15 10:00	4/15 10:20	.33 hours

Lab 1 Writeup

NR2	Tyler	4/15 10:20	4/15 11:50	.5 hours
Parsing teachers	Evin	4/15 10:00	4/15 10:30	.5 hours
NR3	Evin	4/15 10:30	4/15 11:00	.5 hours
Teacher analysis	Tyler	4/17 10:40	4/17 11:00	.25 hours
NR4 and grade analysis	Evin	4/17 10:40	4/17 11:00	.25 hours

Testing:

During testing, we encountered some bugs with incorrectly naming objects which resulted in the Teacher list not being filled. We also had a bug where we didn't cast the Teacher's classroom to an int when stored, so on search for a classroom, the given classroom number didn't match the data type of the Teacher classroom and no matches were found.

We found a bug where the print would only work for certain versions of Python (3.6 and higher). As such, we recommend only using Python 3.6 and higher to ensure the proper output.

Final Thoughts:

Like part 1, the size of the provided student and teacher files is small enough that our implementation of this project works great. Still, it wouldn't scale very well in a situation with many entries, as the entire file needs to be loaded into memory and stored as Student and Teacher objects before any searching can be done. It also doesn't account for error checking on the files, so if there are any errors, data might be loaded wrong. Despite this, we gained a lot of experience with implementing relational queries and a better understanding of the usefulness of different data structures depending on the requirements.

-

Test Suite for Lab 1, Part 1

```
// TC-1
// Tests Requirements R3, R4
// Short form command name, existing student
// Expected output: "  SARAO, DIEDRA, 2, 108, HAMER, GAVIN"
```

S SARAO

```
// TC-2
// Tests Requirements R3, R4
```

Lab 1 Writeup

```
// Short form command name, non-existing student
// Expected output: None
```

S FALANTHROPO

```
// TC-3
// Tests Requirements R3, R5
// Short form command name, existing student with bus route
// Expected output: "  SARAO, DIEDRA, 52"
```

S SARAO B

```
// TC-4
// Tests Requirements R3, R5
// Short form command name, non-existing student with bus route
// Expected output: None
```

S MARFAH B

```
// TC-5
// Tests Requirements R3, R6
// Short form command name, existing TEACHER
// Expected output: "Students belonging to teacher:
                    COOKUS,XUAN
                    ELHADDAD,SHANTE
                    SWEDLUND,SHARRI
                    CIGANEK,MANIE
                    COVINGTON,TOMAS
                    EARLY,TORY
                    LINHART,LELA"
```

T FAFARD

```
// TC-6
// Tests Requirements R3, R6
// Short form command name, non-existing teacher
// Expected output: "Students belonging to teacher:"
```

T AYERON

```
// TC-7
// Tests Requirements R3, R7, NR3
// Short form command name, grade number
// Expected output: "Students in grade level 1
                    SAELEE,DANILO
                    GARTH,JOHN
Teachers whom teach grade level 1
```

Lab 1 Writeup

```
FALKER, ADOLPH"
```

```
G 1
```

```
// TC-8
// Tests Requirements R3, R9
// Short form command name, grade number with low option
// Expected output: "Student with lowest gpa in grade level 1
                    SAELEE,DANILO,2.85,FALKER,ADOLPH,54"
```

```
G 1 L
```

```
// TC-9
// Tests Requirements R3, R9
// Short form command name, grade number with high option
// Expected output: "Student with highest gpa in grade level 1
                    GARTH,JOHN,3.14,FALKER,ADOLPH,0"
```

```
G 1 H
```

```
// TC-10
// Tests Requirements R3, R8
// Short form command name, bus route
// Expected output: "Students who take bus 0
                    SCHOENECKER,PHUONG,6,109
                    FINCHMAN,MATHILDA,6,111
                    BRODERSEN,HYE,3,110
                    HAVIR,BOBBIE,2,108
                    MASSART,ELDON,4,105
                    GARTH,JOHN,1,103
                    CREMEANS,RANDOLPH,6,109
                    KREESE,CARRIE,6,109"
```

```
B 0
```

```
// TC-11
// Tests Requirements R3, R10
// Short form command name, average GPA
// Expected output: "   grade level: 1
                    gpa average: 2.995"
```

```
A 1
```

```
// TC-12
// Tests Requirements R3, R11
// Short form command name, grade info
// Expected output: "   Grade 0: 0 Students
```

Lab 1 Writeup

```
Grade 1: 2 Students
Grade 2: 13 Students
Grade 3: 9 Students
Grade 4: 15 Students
Grade 5: 0 Students
Grade 6: 21 Students"
```

I

```
// TC-13
// Tests Requirements E1
// Invalid command
// Expected output: "Invalid or incomplete command"
```

HELP

```
// TC-14
// Tests Requirements R3, R12
// Short form command name, quit
// Expected output: None
```

Q

```
// TC-15
// Tests Requirements NR1, NR2
// Short form for command name, classroom
// Expected output: "  STUDENTS
                    RACANELLO, NOEL
                    CORONADO, DIMPLE
                    BOYTER, WAN
                    KEMERER, ROSETTA
                    DEMARTINI, DEWAYNE

                    TEACHERS
                    KERBS , BENITO"
```

C

```
// TC-16
// Tests Requirements NR4
// Short form for command name, enrollment
// Expected output: "Enrollment by classroom:
                    Room: 101, Students: 1
                    Room: 102, Students: 5
                    Room: 103, Students: 2
                    Room: 104, Students: 2
                    Room: 105, Students: 6"
```

Lab 1 Writeup

```
Room: 106, Students: 2
Room: 107, Students: 7
Room: 108, Students: 11
Room: 109, Students: 5
Room: 110, Students: 2
Room: 111, Students: 9
Room: 112, Students: 8

"

E

// TC-17
// Tests Requirements for analysis
// Short form for command name, Analyze grade level
// Expected output: "Average gpa by grade:
    Grade: 1, Average gpa: 2.995
    Grade: 2, Average gpa: 2.946153846153846
    Grade: 3, Average gpa: 3.048888888888889
    Grade: 4, Average gpa: 2.951333333333334
    Grade: 5, Average gpa: 0
    Grade: 6, Average gpa: 2.9771428571428573

"

An G

// TC-18
// Tests Requirements for analysis
// Short form for command name, Analyze teacher's average gpa
// Expected output: "COOL, REUBEN avg gpa: 2.91
    KERBS, BENITO avg gpa: 2.98
    FALKER, ADOLPH avg gpa: 3.00
    STEIB, GALE avg gpa: 2.90
    HANTZ, JED avg gpa: 2.91
    BODZIONY, LUZ avg gpa: 3.09
    FAFARD, ROCIO avg gpa: 3.01
    HAMER, GAVIN avg gpa: 2.95
    GAMBREL, JAE avg gpa: 2.96
    ALPERT, JONATHAN avg gpa: 3.17
    NISTENDIRK, NANCY avg gpa: 2.96
    CHIONCHIO, PERLA avg gpa: 2.98

"

An T

// TC-19
// Tests Requirements for analysis
// Short form for command name, Analyze bus route's average gpa
```

Lab 1 Writeup

```
// Expected output: "GPA by bus route:
    Bus: 0, Average gpa: 2.9525
    Bus: 51, Average gpa: 3.02
    Bus: 52, Average gpa: 2.885
    Bus: 53, Average gpa: 3.0555555555555554
    Bus: 54, Average gpa: 2.9416666666666667
    Bus: 55, Average gpa: 3.0366666666666666
    Bus: 56, Average gpa: 2.9216666666666664
```

"

An B