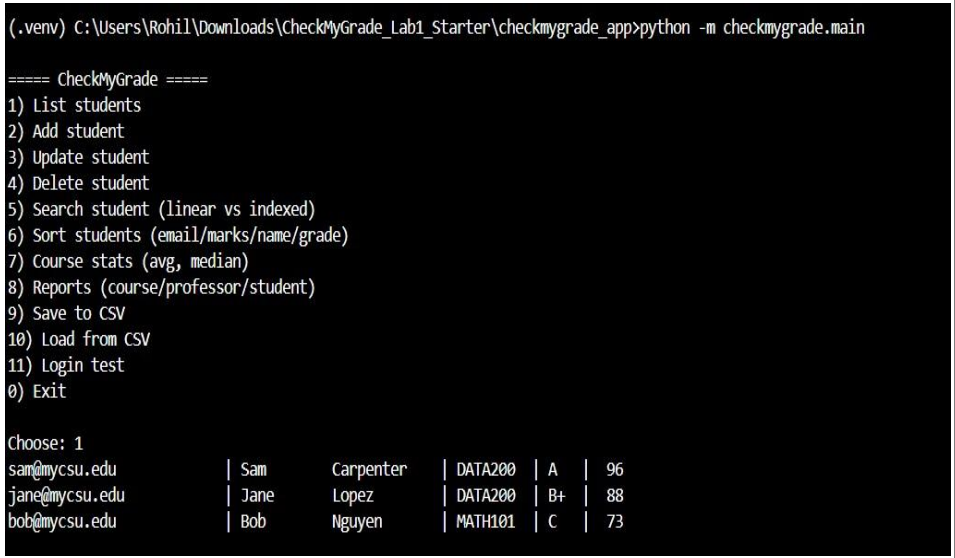


DATA 200 LAB-1

Rohil Utture

Github link: https://github.com/rohiluttire/CheckMyGrade_Lab1

Outputs

Step	Menu Option	Description	Screenshot
1	List Students	Displays all current student records.	 <pre>(.venv) C:\Users\Rohil\Downloads\CheckMyGrade_Lab1_Starter\checkmygrade_app>python -m checkmygrade.main ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 1 sam@mycsu.edu Sam Carpenter DATA200 A 96 jane@mycsu.edu Jane Lopez DATA200 B+ 88 bob@mycsu.edu Bob Nguyen MATH101 C 73</pre>

2	Add Student	Adds a new student record with email, name, course, and marks.	<pre> ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 2 Email: test@sjsu.edu First name: Test Last name: Student Course id: DATA200 Marks (0-100): 82 Added. </pre>
3	Update Student	Updates a student's marks or other fields.	<pre> ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 3 Student email to update: test@sjsu.edu Field (first_name,last_name,course_id,marks): marks New value: 91 Updated. </pre>

4	Delete Student	Deletes a student record by email.	<pre> ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 4 Student email to delete: test@sjsu.edu Deleted. </pre>
5	Search Student	Compares linear vs indexed search with timing result	<pre> ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 5 Student email to search: sam@mycsu.edu Linear search: FOUND in 0.004 ms Indexed search: FOUND in 0.004 ms </pre>

6	Sort Students	Sorts records by email, marks, name, or grade and prints sort timing.	<pre>===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 6 Sort by (email/marks/name/grade): marks Ascending? (y/n): n Sorted in 0.007 ms</pre>
---	------------------	--	---

7	Course Statistics	Shows average and median marks for a selected course.	<pre> ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 7 Course id: DATA200 {'count': 2, 'average': 92.0, 'median': 92.0} </pre>
8a	Course Report	Displays students in a specific course.	<pre> ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 8 Report (course/professor/student): course Course id: DATA200 Email Name Course Grd Mk ----- sam@mycsu.edu Sam Carpenter DATA200 A 96 jane@mycsu.edu Jane Lopez DATA200 B+ 88 </pre>

8b	Professor Report	Displays students taught by a specific professor.	<pre> ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 8 Report (course/professor/student): professor Professor id (email): micheal@mycsu.edu Email Name Course Grd Mk ----- sam@mycsu.edu Sam Carpenter DATA200 A 96 jane@mycsu.edu Jane Lopez DATA200 B+ 88 </pre>
8c	Student Report	Displays report for one specific student.	<pre> ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 8 Report (course/professor/student): student Student email: sam@mycsu.edu Email Name Course Grd Mk ----- sam@mycsu.edu Sam Carpenter DATA200 A 96 </pre>

9	Save to CSV	Saves all current records to CSV files.	<pre> ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 9 Saved. </pre>
10	Load from CSV	Loads data from CSV files into memory.	<pre> ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 10 Loaded. </pre>

11	Login Test	Verifies login using encrypted password from file.	<pre> ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 11 Email: micheal@mycsu.edu Password: welcome12#_ Login OK </pre>
0	Exit	Exits the application.	<pre> ===== CheckMyGrade ===== 1) List students 2) Add student 3) Update student 4) Delete student 5) Search student (linear vs indexed) 6) Sort students (email/marks/name/grade) 7) Course stats (avg, median) 8) Reports (course/professor/student) 9) Save to CSV 10) Load from CSV 11) Login test 0) Exit Choose: 0 Bye </pre>

		students.csv	<pre>Email_address,First_name,Last_name,Course.id,grades,Marks sam@mycsu.edu,Sam,Carpenter,DATA200,A,96 jane@mycsu.edu,Jane,Lopez,DATA200,B+,88 bob@mycsu.edu,Bob,Nguyen,MATH101,C,73</pre>
		Unit Test	<pre>(.venv) C:\Users\Rohil\Downloads\CheckMyGrade_Lab1_Starter\checkmygrade_app>python -m unittest discover -s tests -v test_course_professor_crud (test_app.CheckMyGradeTests.test_course_professor_crud) ... ok test_encryption (test_app.CheckMyGradeTests.test_encryption) ... ok test_search_timing (test_app.CheckMyGradeTests.test_search_timing) ... --- Timing: Search (linear vs indexed) --- Target: jamie0@jsu.edu Linear: 0.001 ms Indexed: 0.001 ms ok test_sorting (test_app.CheckMyGradeTests.test_sorting) ... ok test_stats_and_reports (test_app.CheckMyGradeTests.test_stats_and_reports) ... ok test_student_crud (test_app.CheckMyGradeTests.test_student_crud) ... ok ----- Ran 6 tests in 0.350s OK</pre>

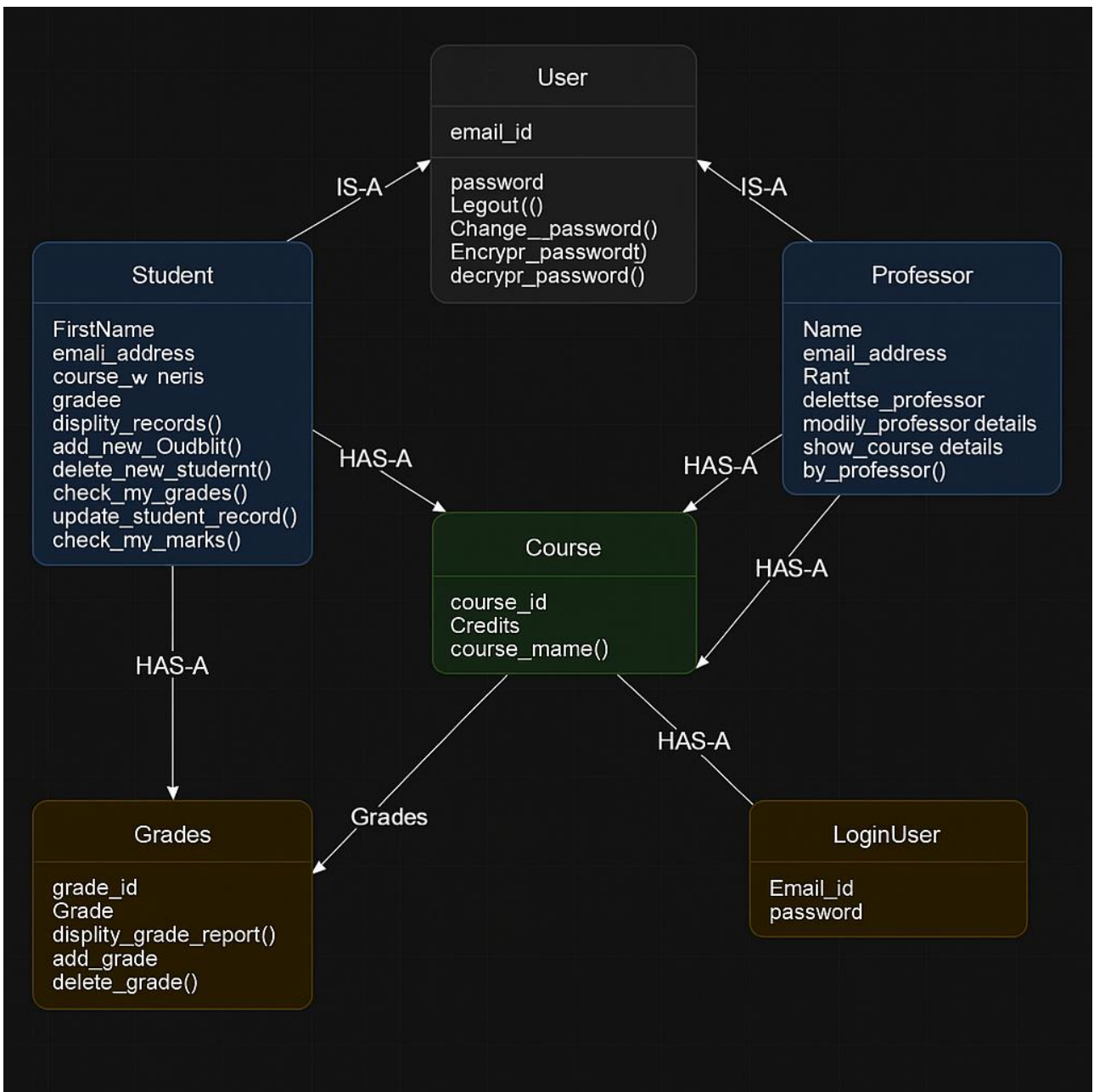


Figure 1:Diagram showing the IS/HAS-A relationships