**Assignment 1**

**(7 Bonus Points)**

**Write your answers in this document itself and upload it to Gradescope**

NAME Adrian Knight

ID 677179394

**Question 1 (5 Points)**

department(dept\_name, building, budget)

instructor(ID, name, dept\_name, salary)

Consider the foreign-key constraint from the dept\_name attribute of **instructor** to the **department** relation. Give examples of inserts and deletes to these relations that can cause a violation of the foreign-key constraint.

Department

|  |  |  |
| --- | --- | --- |
| dept\_name | Building | Budget |
| Science | Building Science | 5 |
| Math | Building Math | 10 |

Instructor

|  |  |  |  |
| --- | --- | --- | --- |
| ID | name | dept\_name | Salary |
| 1 | John | Science | 100 |
| 2 | Mary | Math | 200 |

1. INSERT into instructor values (3, Connor, Chemistry, 150) # Chemistry violates the foreign-key constraint as it does not exist
2. INSERT into department values (Math, Newer Math Building, 30) # Math cannot be the primary key as the key already exists for the tuple with a budget of 10
3. DELETE from department (Science, Building Science, 5) # Deleting the science tuple results in Instructor with ID = 1 having a foreign key that points to nothing.

**Question 2**

**Student**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Student\_ID | Name | Age | Gender | Major | Student\_type | GPA | Advisor\_name |
| 670190001 | Oliver Hayes | 21 | F | CS | Undergraduate | 3.5 | Thomas Bradley |
| 670190002 | Liam Bennett | 24 | M | CS | Master | 3.8 | David Foster |
| 670190003 | Ava Turner | 22 | M | Math | Undergraduate | 3.2 | Robert Sinclair |
| 670190004 | Ethan Mitchell | 30 | M | CS | PhD | 3.8 | David Foster |
| 670190005 | Mason Brooks | 26 | F | Biology | PhD | 4.0 | Olivia Coleman |
| 670190006 | Lucas Price | 20 | M | EE | Undergraduate | 3.3 | Andrew Bennett |
| 670190007 | Mia Hughes | 21 | F | Nursing | Undergraduate | 3.7 | Sophia Caldwell |
| 670190008 | Henry Russell | 23 | M | Math | Master | 3.5 | James Whitaker |
| 670190009 | Emily Jenkins | 25 | M | Physics | Master | 3.5 | Gregory Winters |
| 670190010 | Elijah Torres | 22 | M | Chemistry | Undergraduate | 3.6 | Abigail Mercer |
| 670190011 | James Carter | 27 | F | Biology | PhD | 3.7 | Olivia Coleman |
| 670190012 | Sophia Evans | 24 | F | EE | Master | 3.1 | Marcus Caldwell |
| 670190013 | Noah Bennett | 26 | M | CS | PhD | 4.0 | William Thornton |
| 670190014 | Mia Scott | 22 | M | Biology | Undergraduate | 3.4 | Lydia Keaton |

For the above relation named ***Student***, what is the output of the following four queries: **(5 x 5 = 25 points)**

|  |  |
| --- | --- |
| Student\_ID | Name |
| 670190002 | Liam Bennett |
| 670190004 | Ethan Mitchell |
| 670190005 | Mason Brooks |
| 670190007 | Mia Hughes |
| 670190011 | James Carter |
| 670190010 | Elijah Torres |
| 670190013 | Noah Bennett |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Student\_ID | Name | | Major | | |
| 670190001 | | | Oliver Hayes | | CS |
| 670190002 | | | Liam Bennett | | CS |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Student\_ID | Name | | Major | | |
| 670190001 | | Oliver Hayes | | CS |
| 670190002 | | Liam Bennett | | CS |
| 670190003 | | Ava Turner | | Math |
| 670190004 | | Ethan Mitchell | | CS |
| 670190006 | | Lucas Price | | EE |
| 670190008 | | Henry Russell | | Math |
| 670190009 | | Emily Jenkins | | Physics |
| 670190010 | | Elijah Torres | | Chemistry |
| 670190013 | | Noah Bennett | | CS |
| 670190014 | | Mia Scott | | Biology |



|  |
| --- |
| Student\_ID |

670190005

670190011



|  |
| --- |
| Major |
| CS |
| CS |
| Math |
| CS |
| Biology |
| EE |
| Nursing |
| Math |
| Physics |
| Chemistry |
| Biology |
| EE |
| CS |
| Biology |

**Question 3**

Consider a database with the following schema:

* Students (student\_id, name, major, GPA )
* Instructors (instructor\_id, name, department, course\_id)
* Courses ( course\_id, name, instructor, credits)
* Enrollments ( student\_id, course\_id, instructor\_id, grade)

**Student Instructors**

|  |  |  |  |
| --- | --- | --- | --- |
| student\_id | name | major | GPA |
| 670190001 | Oliver Hayes | CS | 3.5 |
| 670190002 | Liam Bennett | CS | 3.8 |
| 670190003 | Ava Turner | Math | 3.2 |
| 670190004 | Ethan Mitchell | CS | 3.8 |
| 670190005 | Mason Brooks | Biology | 4.0 |
| 670190006 | Lucas Price | EE | 3.3 |
| 670190007 | Mia Hughes | Nursing | 3.7 |

|  |  |  |  |
| --- | --- | --- | --- |
| instructor\_id | name | department | course\_id |
| UIC0010 | Thomas Bradley | CS | 2403 |
| UIC0014 | David Foster | CS | 2406 |
| UIC0023 | Robert Sinclair | Math | 3401 |
| UIC0037 | Olivia Coleman | Biology | 0511 |
| UIC0040 | Andrew Bennett | EE | 2511 |

**Courses Enrollments**



|  |  |  |  |
| --- | --- | --- | --- |
| course\_id | name | instructor | credits |
| 2403 | databases | Thomas Bradley | 4 |
| 2406 | Machine Learning | David Foster | 4 |
| 3401 | Linear Algebra | Robert Sinclair | 5 |
| 0511 | Cell Biology | Olivia Coleman | 3 |
| 2511 | Control Systems | Andrew Bennett | 5 |

| student\_id | courses\_id | instructor\_id | grade |
| --- | --- | --- | --- |
| 670190001 | 2403 | UIC0010 | A |
| 670190002 | 2403 | UIC0010 | B |
| 670190003 | 3401 | UIC0023 | A |
| 670190004 | 2406 | UIC0014 | A |
| 670190005 | 0511 | UIC0037 | B |
| 670190006 | 2511 | UIC0037 | B |
| 670190007 | 0511 | UIC0037 | A |

**Write query in relational algebra for the following (5 x 5 = 25 Points)**

1. Find the names of all students enrolled in the course **"databases".**



1. Find the names of instructors teaching students with a **GPA of 4.0**.

(student enrollments instructors)



1. Find the names of students and the courses they are enrolled in along with the instructor's name for those who received an **A grade**.

student.name, courses.instructor, courses.name ( grade=”A” (student enrollments courses))



1. List the names of instructors who are teaching students majoring in **"CS"**.
2. Find the names of students with a **GPA greater** **than 3.5** and enrolled in courses taught by **"Robert Sinclair"**.

**Question 4 (8 x 4 = 32 Points)**

Employee database:

employee (ID, person\_name, street, city)

company (company\_name, city)

Works (ID, company\_name, salary) → ID is a foreign key referencing the ID of employee, and company\_name is a foreign key referencing the company\_name of the company table.

For the above schema of the employee database. Give an expression in the relational algebra to express each of the following queries:

1. Find the name of each employee living in Chicago.
2. Find the name of each employee whose salary exceeds $50000.
3. Find the name of each employee who lives in Chicago and whose salary is greater than $50000.
4. Find the ID and name of each employee who does not work for BigBank.
5. Find the ID and name of each employee who works for “BigBank”.
6. Find the ID, name, and city of residence of each employee who works for “BigBank”.
7. Find the ID, name, street address, and city of residence of each employee who works for “BigBank” and earns more than $10000.
8. Find the ID and name of each employee in this database who lives in the same city as the company for which she or he works.

**Question 5 (10 Points)**

Write a relational expression to compute all paths of length 3 for any given directed graph G. Show the working of your RA expression on a real graph of 7 nodes and 7 edges (use your imagination to create the graph)

**Question 6 (10 Points)**

instructor(ID, name, dept\_name, salary)

Write a relational algebra query for the above table that computes the maximum salary.