

Project Part 2

a.

- i. **Department** (department_name, chair_name, num_of_faculty)
 - 1. Primary Key: department_name
- ii. **Hosting** (department_name, event_name)
 - 1. Primary Key: department_name, event_name
 - 2. Foreign Key: department_name references Department(department_name) & event_name references Event(event_name)
- iii. **Major** (major_code, major_name, department_name)
 - 1. Primary Key: major_code
 - 2. Alternate Key: major_name
 - 3. Foreign Key: department_name references Department(department_name)
- iv. **Declaring** (major_code, stu_id)
 - 1. Primary Key: major_code, stu_id
 - 2. Foreign Key: major_code references Major(major_code) & stu_id references Student(stu_id)
- v. **Student** (stu_id, stu_name, stu_initials)
 - 1. Primary Key: stu_id
- vi. **Attending** (stu_id, event_num)
 - 1. Primary Key: stu_id, event_num
 - 2. Foreign Key: stu_id references Student(stu_id) & event_num references Event(event_num)
- vii. **Event** (event_num, event_name, start_date, end_date)
 - 1. Primary Key: event_num

b.

- i. Dependencies:
 - 1. department_name → chair_name, num_of_faculty (primary key)
 - 2. major_code → major_name, department_name (primary key)
 - 3. stu_id → stu_name, stu_initials (primary key)

4. event_num → event_name, start_date, end_date, department_name
(primary key)

a. 1NF:

- i. **Department** (department_name, chair_name, num_of_faculty)
- ii. **Hosting** (department_name, event_name)
- iii. **Major** (major_code, major_name, department_name)
- iv. **Declaring** (major_code, stu_id)
- v. **Student** (stu_id, stu_name, stu_initials)
- vi. **Attending** (stu_id, event_num)
- vii. **Event** (event_num, event_name, start_date, end_date)

b. 2NF: No partial dependencies so already in 2NF

- i. **Department** (department_name, chair_name, num_of_faculty)
- ii. **Hosting** (department_name, event_name)
- iii. **Major** (major_code, major_name, department_name)
- iv. **Declaring** (major_code, stu_id)
- v. **Student** (stu_id, stu_name, stu_initials)
- vi. **Attending** (stu_id, event_num)
- vii. **Event** (event_num, event_name, start_date, end_date)

c. 3NF: No transitive dependencies so already in 3NF

- i. **Department** (department_name, chair_name, num_of_faculty)
- ii. **Hosting** (department_name, event_name)
- iii. **Major** (major_code, major_name, department_name)
- iv. **Declaring** (major_code, stu_id)
- v. **Student** (stu_id, stu_name, stu_initials)
- vi. **Attending** (stu_id, event_num)
- vii. **Event** (event_num, event_name, start_date, end_date)

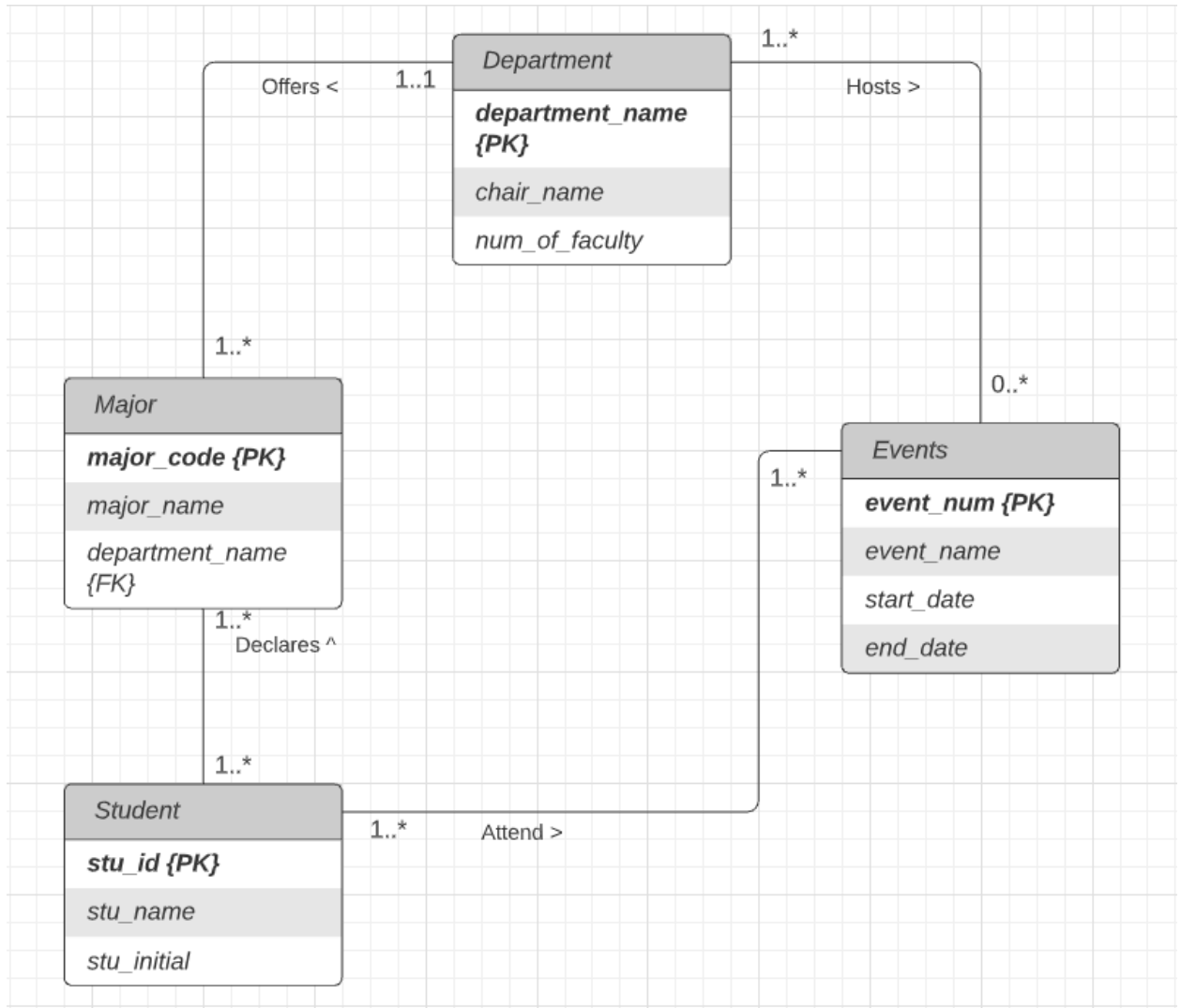
c.

- i. List all events that the student with stu_id 001 attended.

1. We would do this by accessing the Attending table and selecting the event_num where the stu_id is 001. If we wanted the event_name of this, we would look in the Event table and match the event_num with the name.
- ii. List the number of departments that host each event.
 1. We would do this by accessing the Hosting table and grouping the event_num and counting the number of departments each event_num is associated with.
- iii. How many faculty members are there in the University?
 1. We can do this by taking the sum of the num_of_faculty of each department.
- iv. List all the chair_names of the department of each major.
 1. We can do this by seeing which department each major belongs to, and then going to the Department table to find the corresponding chair_name.
- v. List the number of students that have declared each major.
 1. We can do this by looking in the Declaring table. We can group by the major_code and count the number of times we see that major_code.
- vi. List the name of the major that each student has declared.
 1. We could do this by looking into the Declaring table and referring to the major_code in the Major table in order to get the major_name to go with each entry in the Declaring table.
- vii. List the event_name, department_name, and chair of department for all events that are scheduled to start after Dec 1.
 1. We can do this by looking in the Event table to see which events are starting after Dec 1. After this we can look in the Hosting table to get the department name for each event. Then we can look in the Department table to get the name of the chair.
- d. Under each table, I have listed the primary key, foreign key, and alternate key and their constraints if they apply.
 - i. **Department** (department_name, chair_name, num_of_faculty)
 1. PK: department_name

- a. Constraint: Must be UNIQUE and NOT NULL
 - 2. FK: None
 - 3. AK: None
- ii. **Hosting** (department_name, event_name)
 - 1. PK: department_name, event_name
 - a. Constraint: Must be UNIQUE and NOT NULL
 - 2. FK:
 - a. department_name references Department(department_name)
 - b. event_name references Event(event_name)
 - i. Constraint: Each foreign key must have a match in the referring table or all attributes a part of the FK must be NULL
 - 3. AK: None
- iii. **Major** (major_code, major_name, department_name)
 - 1. PK: major_code
 - a. Constraint: Must be UNIQUE and NOT NULL
 - 2. FK: department_name
 - a. Constraint: Each foreign key must have a match in the referring table or all attributes a part of the FK must be NULL
 - 3. AK: major_name
 - a. Constraint: Must be UNIQUE and NOT NULL
- iv. **Declaring** (major_code, stu_id)
 - 1. PK: major_code, stu_id
 - a. Constraint: Must be UNIQUE and NOT NULL
 - 2. FK:
 - a. major_code references Major(major_code)
 - b. stu_id references Student(stu_id)
 - i. Constraint: Each foreign key must have a match in the referring table or all attributes a part of the FK must be NULL
 - 3. AK: None

- v. **Student** (stu_id, stu_name, stu_initials)
 - 1. PK: stu_id
 - a. Constraint: Must be UNIQUE and NOT NULL
 - 2. FK: None
 - 3. AK: None
- vi. **Attending** (stu_id, event_num)
 - 1. PK: stu_id, event_num
 - a. Constraint: Must be UNIQUE and NOT NULL
 - 2. FK:
 - a. stu_id references Student(stu_id)
 - b. event_num references Event(event_num)
 - i. Constraint: Each foreign key must have a match in the referring table or all attributes a part of the FK must be NULL
 - 3. AK: None
- vii. **Event** (event_num, event_name, start_date, end_date)
 - 1. PK: event_num
 - a. Constraint: Must be UNIQUE and NOT NULL
 - 2. FK: None
 - 3. AK: {event_name, start_date}
 - a. Constraint: Must be UNIQUE and NOT NULL
- viii. General Constraints:
 - 1. end_date must be after start_date
 - 2. start_date must not be a past or the current date; it must be a future date
 - 3. major_code must be 3 characters
 - 4. stu_initial must be more than one character



e.