

# CSC343 Worksheet 5

June 18, 2020

1. **Exercise 7.1.1:** Our running example movie database of Section 2.2.8 has keys defined for all its relations.

```
1  Movies(title, year, length, genre, studioName, producerC#)
2  StarsIn(movieTitle, movieYear, starName)
3  MovieStar(name, address, gender, birthdate)
4  MovieExec(name, address, cert#, netWorth)
5  Studio(name, address, presC#)
6
```

Declare the following referential integrity constraints for the movie database as in Exercise 7.1.1.

- a) The producer of a movie must be someone mentioned in MovieExec. Modifications to MovieExec that violate this constraint are rejected.
  - b) Repeat (a), but violations result in the producerC# in Movie being set to NULL.
  - c) Repeat (a), but violations result in the deletion or update of the offending Movie tuple.
  - d) A movie that appears in Stars In must also appear in Movie. Handle violations by rejecting the modification.
  - e) A star appearing in Stars In must also appear in MovieStar. Handle violations by deleting violating tuples.
2. **Exercise 7.1.2:** We would like to declare the constraint that every movie in the relation Movie must appear with at least one star in StarsIn. Can we do so with a foreign-key constraint? Why or why not?
  3. **Exercise 7.1.3:** Suggest suitable keys and foreign keys for the relations of the PC database:

```
1  Product(maker, model, type)
2  PC(model, speed, ram, hd, price)
3  Laptop(model, speed, ram, hd, screen, price)
4  Printer(model, color, type, price)
5
```

of Exercise 2.4.1. Modify your SQL schema from Exercise 2.3.1 to include declarations of these keys.

4. **Exercise 7.1.4:** Suggest suitable keys for the relations of the battleships database

```
1  Classes(class, type, country, numGuns, bore, displacement)
2  Ships(name, class, launched)
3  Battles(name, date)
4  Outcomes(ship, battle, result)
5
```

of Exercise 2.4.3. Modify your SQL schema from Exercise 2.3.2 to include declarations of these keys.

5. **Exercise 7.1.5:** Exercise 7.1.5: Write the following referential integrity constraints for the battleships database as in Exercise 7.1.4. Use your assumptions about keys from that exercise, and handle all violations by setting the referencing attribute value to NULL

- a) Every class mentioned in **Ships** must be mentioned in **Classes**.
- b) Every battle mentioned in **Outcomes** must be mentioned in **Battles**.
- c) Every ship mentioned in **Outcomes** must be mentioned in **Ships**.