Answers

1. What is the cardinality of relation Movies? **7**

2. What is the arity of relation Artists? **3**

3. What is the key of relation Movies? **mID**

4. Change the instance of relation Artists in a way that violates its key constraint. Okay, you can unchange now. `:-)`

   There are **many answers**. One is to change the aID of Fisher to 2.

   Artists

aID	aName	nat
1	Nicholson	American
2	Ford	American
3	Stone	British
2	Fisher	American

5. According to the schema, is there any limit on the number of directors a movie can have? **Yes, one**

   If we allowed two names to go into a single cell, then we could record two directors for a movie. But if not, in order to record the second director, we'd have to add a new tuple that repeats the movie's mID and put the second director there. But this is not allowed because mID is a key for relation Movies.

6. According to the schema, can exactly the same movie title be used for two different movies? **Yes**

   This is fine *as long as* we use a different mID.

7. Change the instance of the database so that the constraint Roles[aID] $\subseteq$ Artists[aID] is violated.

   There are **many ways** to do this. we could change the aID for Han Solo in Roles to 99, or we could remove the last row from Artists, or we could change the aID for Stone to 55, for example.

8. Does any actor show up in relation Roles twice with the same mID?

   Not in the given instance

9. According to the schema, *can* an actor show up in relation Roles twice with same mID?

   Yes. Nothing prevents an instance from including an actor in Roles twice with the same mID. Think about what this would represent in the world of movies.