

Homework 1

CS5800

Jeremy Desser

October 1, 2015

1 Relational Algebra

1.1

$\sigma_{AGE} Actor$

1.2

$\pi_{Title}(\sigma_{WhenReleased="1940"} Movie)$

1.3

1. $\pi_{Title}(\sigma_{Cost>1000000 \wedge WhenReleased<"1920"} Movie)$
2. $Movies_{Title} \cap Movies_{cost>1000000 \wedge WhenReleased<"1920"}$

1.4

$\pi_{Name, Age}(\sigma_{Movie.WhenReleased="1940"} (Actor \bowtie (CastIn \bowtie Movie)))$

1.5

$\pi_{Name, Age}(\sigma_{Movie.WhenReleased<"1920"} (Actor \bowtie (CastIn \bowtie Movie)))$

1.6

$(\sigma_{Name} Actor) - (\pi_{Actor.Name}(\sigma_{CastIn.Name} (Actor \bowtie (CastIn \bowtie Movie))))$

1.7

$\pi_{Name}(\sigma_{R_1.Name = R_2.Name}(\rho(R_1, (Actor \bowtie CastIn)) \bowtie \rho(R_2, (Actor \bowtie CastIn))))$

1.8

$\pi_{Name}((CastIn \bowtie (Movie \bowtie Actor)))$

1.9

$\rho_{R_1}(Actor \bowtie CastIn \sigma_{WhenReleased} \bowtie Movie)$
 $\rho_{R_2}(Actor \bowtie CastIn \sigma_{WhenReleased} \bowtie Movie)$
 $\rho_{R_3}(Actor \bowtie CastIn \sigma_{WhenReleased} \bowtie Movie)$
 $\pi_{Name}(\sigma_{R_1.WhenReleased = R_2.WhenReleased+1 \wedge R_2.WhenReleased = R_3+1 \vee R_1.WhenReleased = R_2.WhenReleased-1 \wedge R_2.WhenReleased = R_3.WhenReleased+1} (R_1 \bowtie (R_2 \bowtie R_3)))$

1.10

$(\pi_{Name}(Actor \bowtie (Movie \bowtie CastIn))) \bowtie (\pi_{Name}(\sigma_{R_1.Name = R_2.Name}(\rho(R_1, (Actor \bowtie CastIn)) \bowtie \rho(R_2, (Actor \bowtie CastIn)))) - (\sigma_{Name} Actor) - (\pi_{Actor.Name}(\sigma_{CastIn.Name} (Actor \bowtie (CastIn \bowtie Movie))))$

2 Exercise 2.14

Three-tier architecture, it adds a middle layer that helps keep all the rules managed for the clients. The two-tiered architecture might end up being a bit confusing for someone just looking to buy plane tickets since they would have to deal with api's and such.

3 Exercise 3.12

- a. update Insert and Number_of_available_seats of Leg_Instance with flight_number = xxxxx to Number_of_available_seats = Number_of_available_seats -1.
- b. You would need to check that there are no seats left, the flight date could have passed
- c. The Insert Operation onto the Seat_reservation table will check all the constraints for the relation. number_of_available_seats on each Leg_Instance of the flight < 1 doesn't fall into any of the listed constraints.

4 Exercise 3.14

Order(Cust#) is a foreign key on Customer(Cust#)

Order(Order#) is a foreign key on Order_Item(Order#)

Shipment(Warehouse#) is a foreign key on Warehouse(Warehouse#)

Order(Order#) is a foreign key on Shipment(Order#)