In this assignment you are to write relational algebra queries over a small database.

We've created a small sample database to use for this assignment. It contains four relations:  
Person(name, age, gender) // name is a key  
Frequents(name, pizzeria) // [name,pizzeria] is a key  
Eats(name, pizza) // [name,pizza] is a key  
Serves(pizzeria, pizza, price) // [pizzeria,pizza] is a key

Person

|  |  |  |
| --- | --- | --- |
| **name** | **age** | **gender** |
| Amy | 16 | female |
| Ben | 21 | male |
| Cal | 33 | male |
| Dan | 13 | male |
| Eli | 45 | male |
| Fay | 21 | female |
| Gus | 24 | male |
| Hil | 30 | female |
| Ian | 18 | male |

Frequents

|  |  |
| --- | --- |
| **name** | **pizzeria** |
| Amy | Pizza Hut |
| Ben | Chicago Pizza |
| Ben | Pizza Hut |
| Cal | New York Pizza |
| Cal | Straw Hat |
| Dan | New York Pizza |
| Dan | Straw Hat |
| Eli | Chicago Pizza |
| Eli | Straw Hat |
| Fay | Dominos |
| Fay | Little Caesars |
| Gus | Chicago Pizza |
| Gus | Pizza Hut |
| Hil | Dominos |
| Hil | Pizza Hut |
| Hil | Straw Hat |
| Ian | Dominos |
| Ian | New York Pizza |
| Ian | Straw Hat |

Eats

|  |  |
| --- | --- |
| **name** | **pizza** |
| Amy | mushroom |
| Amy | pepperoni |
| Ben | cheese |
| Ben | pepperoni |
| Cal | supreme |
| Dan | cheese |
| Dan | mushroom |
| Dan | pepperoni |
| Dan | sausage |
| Dan | supreme |
| Eli | cheese |
| Eli | supreme |
| Fay | mushroom |
| Gus | cheese |
| Gus | mushroom |
| Gus | supreme |
| Hil | cheese |
| Hil | supreme |
| Ian | pepperoni |
| Ian | supreme |

Serves

|  |  |  |
| --- | --- | --- |
| **pizzeria** | **pizza** | **price** |
| Chicago Pizza | cheese | 7.75 |
| Chicago Pizza | supreme | 8.5 |
| Dominos | cheese | 9.75 |
| Dominos | mushroom | 11 |
| Little Caesars | cheese | 7 |
| Little Caesars | mushroom | 9.25 |
| Little Caesars | pepperoni | 9.75 |
| Little Caesars | sausage | 9.5 |
| New York Pizza | cheese | 7 |
| New York Pizza | pepperoni | 8 |
| New York Pizza | supreme | 8.5 |
| Pizza Hut | cheese | 9 |
| Pizza Hut | pepperoni | 12 |
| Pizza Hut | sausage | 12 |
| Pizza Hut | supreme | 12 |
| Straw Hat | cheese | 9.25 |
| Straw Hat | pepperoni | 8 |
| Straw Hat | sausage | 9.75 |

Instructions: You have to write relational algebra expressions over the pizza database. We strongly suggest that you work the queries out on paper first, using conventional relational algebra symbols. When you're satisfied with your solution for a given problem, click the "Submit" button.  
(10 Marks for each Question)  
Q1. Find all pizzas eaten by at least one female over the age of 20.   
Q2. Find the names of all females who eat at least one pizza served by Straw Hat. (Note: The pizza need not be eaten at Straw Hat.)   
Q3. Find all pizzerias that serve at least one pizza for less than $10 that either Amy or Fay (or both) eat.   
Q4. Find all pizzerias that serve at least one pizza for less than $10 that both Amy and Fay eat.   
Q5. Find the names of all people who eat at least one pizza served by Dominos but who do not frequent Dominos.

Q6.Find all pizzas that are eaten only by people younger than 24, or that cost less than $10 everywhere they're served.

Q7.Find the age of the oldest person (or people) who eat mushroom pizza.   
*(This query is quite challenging; congratulations if you get it right.)*

problem

1. πEats.pizza (σPerson.gender = ‘female’ ˄ Person.age>20 (Person Naturaljoin Eats))
2. πPerson.name(σFrequent.pizzeria= ‘StrwHat’ ˄ Person.gender= ‘female’ (Person Naturaljoin Frequents))
3. πServes.pizzeria (σServes.price<10 ˄ Eats.name=’Amy’ ˅ Eats.name=’Fay’ (Serves Naturaljoin Eats))
4. πServes.pizzeria (σServes.price<10 ˄ Eats.name=’Amy’ ˄ Eats.name=’Fay’ (Serves Naturaljoin Eats))

6. πServes.pizzeria (Person Naturaljoin person.age<24 Serves Naturaljoin serves.price<10 Frequents)

7.πPerson.age - π P1.age (π P1.age (σ p1.age<p2.age (Þ p1 (Person )Naturaljoin Þ p2 (Person))))