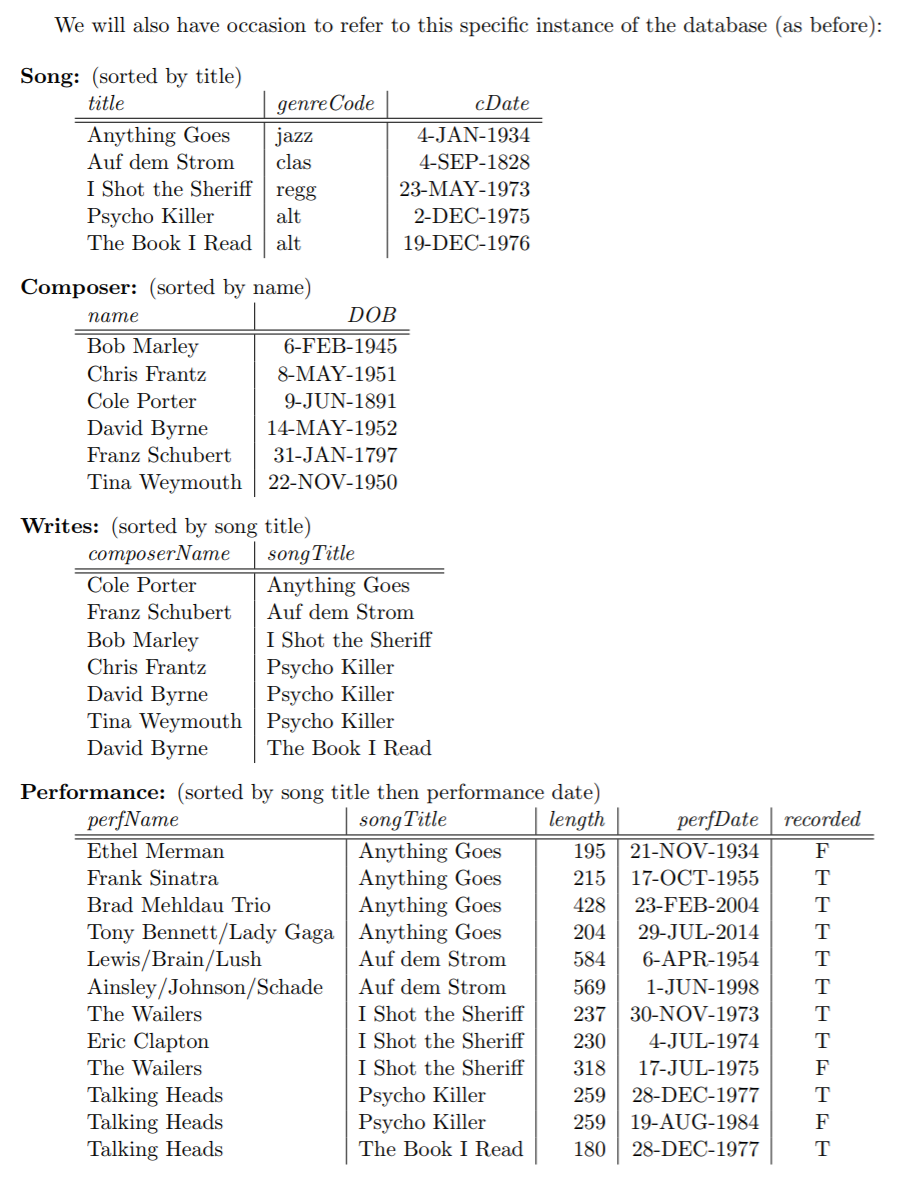
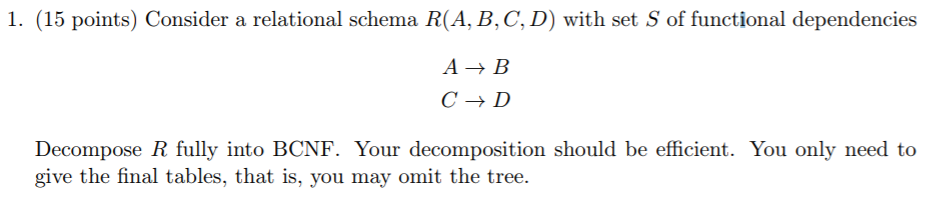


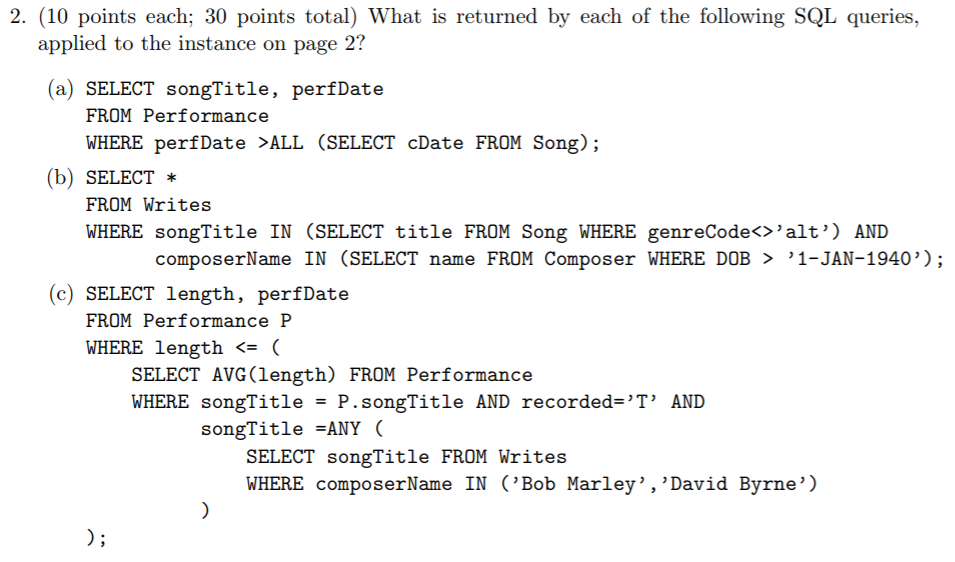
**Fall Midterm 2**







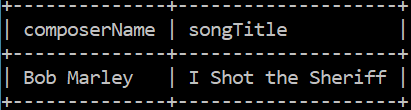




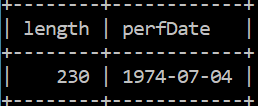
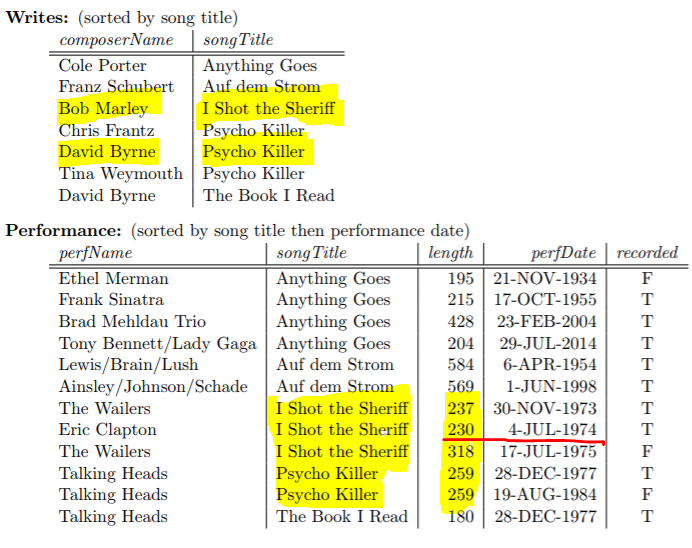
1. Returns all song titles and performance dates of performances that were **performed after** they **the most recent composed song**.

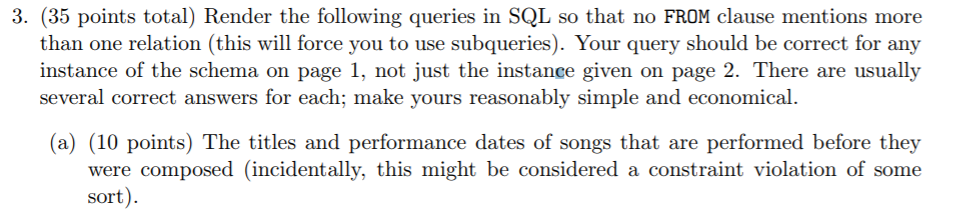


1. Returns all composer names and song titles of all songs **not in the alt genre** and composed by **a composer born after January 1st, 1940**.



1. Returns the length and performance date of the performance **with the shortest length** of a song composed by **Bob Marley or David Byrne**

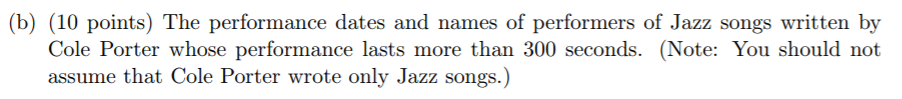
­­

**SELECT** songTitle, perfDate

**FROM** Performance

**WHERE** perfDate **< ALL (SELECT** cDate **FROM** Song**);**





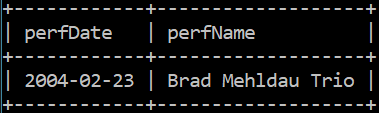
**SELECT** perfDate, perfName

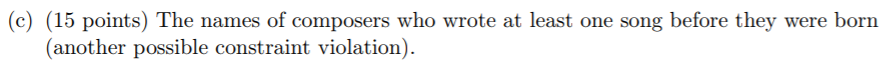
**FROM** Performance

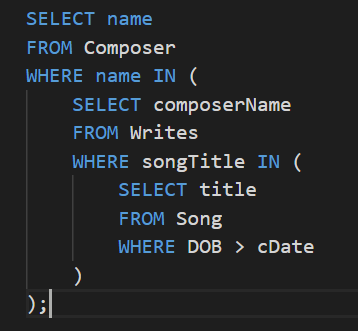
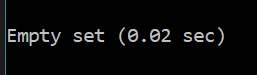
**WHERE** length >= 300 **AND** songTitle **IN (SELECT** songTitle

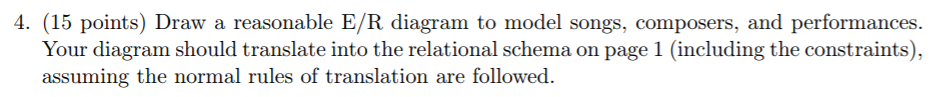
**FROM** Writes

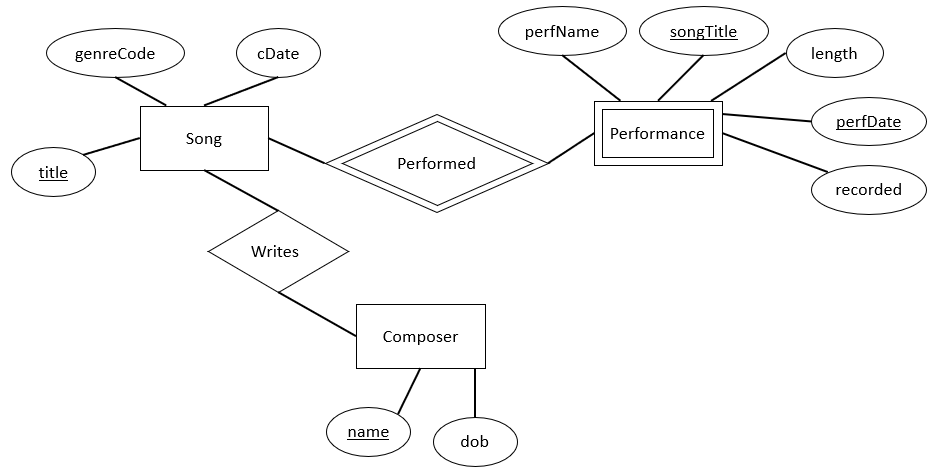
**WHERE** composerName = ‘Cole Porter’**);**

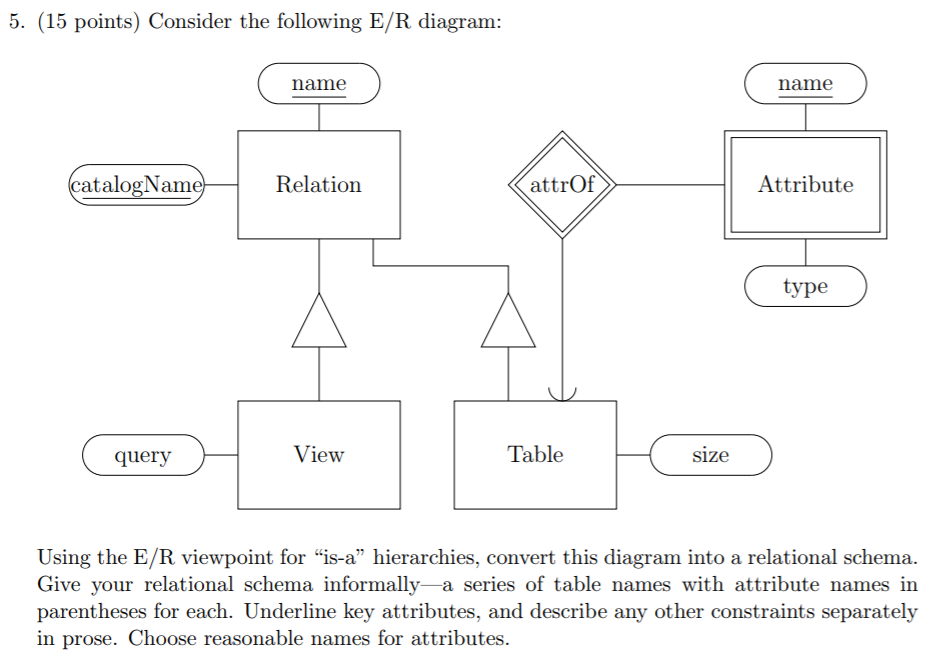










**View (** catalogName, name, query **)**

**Table (** catalogName, name, size )

**Relation (** catalogName, name **)**

**Attribute (** name, type, relationName?, relationCatalogName **)**

**Unfinished UML Diagram**

**Table**

size

**View**



0.. \*

PK

1.. 1

attrOf

**Attribute**

name PK

type

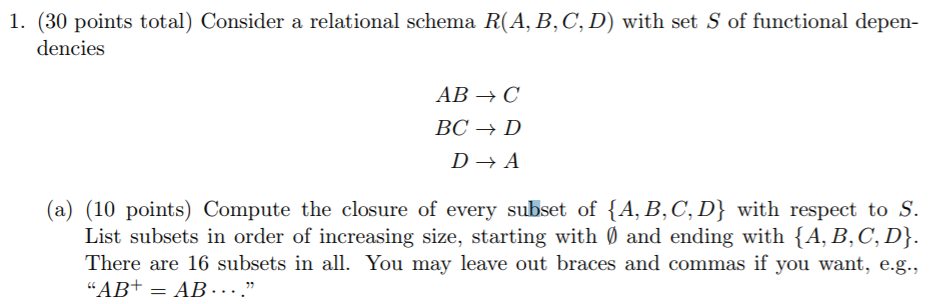
query

catalogName PK

name PK

**Relation**

**Spring Midterm 2**



**There are not 16 subsets, there are 2^4 – 1 = 15 subsets excluding the empty set.**

**{}+ = {}**

**A+ = A**

**B+ = B**

**C+ = C**

**D+ = DA**

**AB+ = ABCD**

**AC+ = AC**

**AD+ = AD**

**BC+ = BCDA**

**BD+ = BDAC**

**CD+ = CDA**

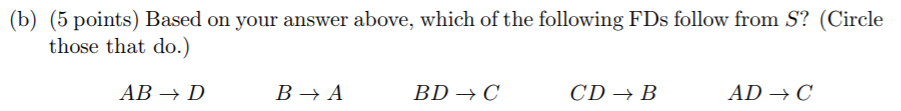
**ABC+ = ABCD**

**ABD+ = ABDC**

**ACD+ = ACD**

**BCD+ = BCDA**

**ABCD+ = ABCD**

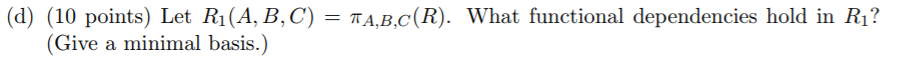


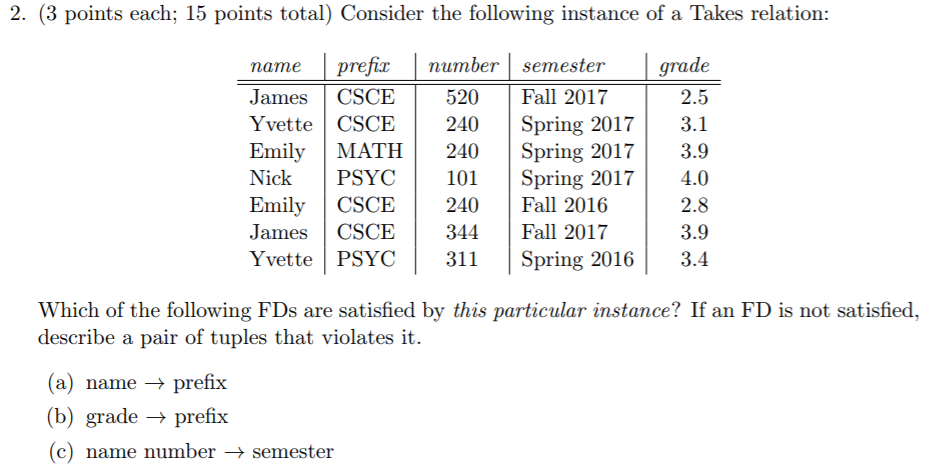


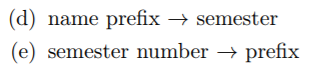


**Keys of R: {AB}, {BC}, {BD}**

**Super keys of R: {ABC}, {ABD}, {BCD}, {ABCD}**







1. **Violation: a name can take more than one class.**

**Ex: Emily MATH and Emily CSCE**

1. **Violation: a specific class is determined by the combined class prefix and number.**

**Ex: CSCE 520 and CSCE 240**

1. **Violation: classes could have the same number in different departments (prefix)**

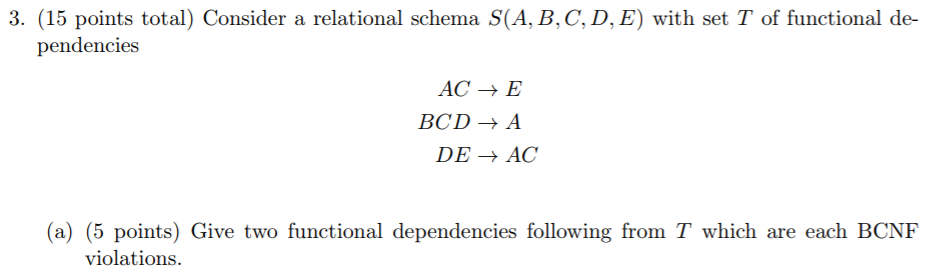
**Ex: Emily (\*MATH) 240 = Spring 2017 and Emily (\*CSCE) 240 = Fall 2016**

1. **No violation for our particular instance. A problem that could occur would be when a student is taking multiple semesters of the same course prefix.**

**Ex: James CSCE = Fall 2017 and James CSCE = Spring 2017**

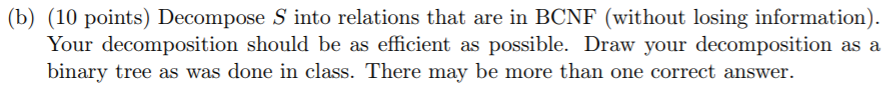
1. **Violation: multiple class numbers (different prefix will determine a different class) during the same semester**

**Ex: Spring 2017 240 = CSCE and Spring 2017 240 = MATH**



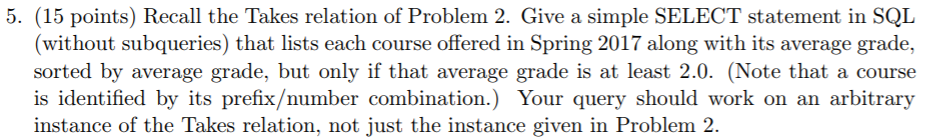
**ABC+ = ABCE**

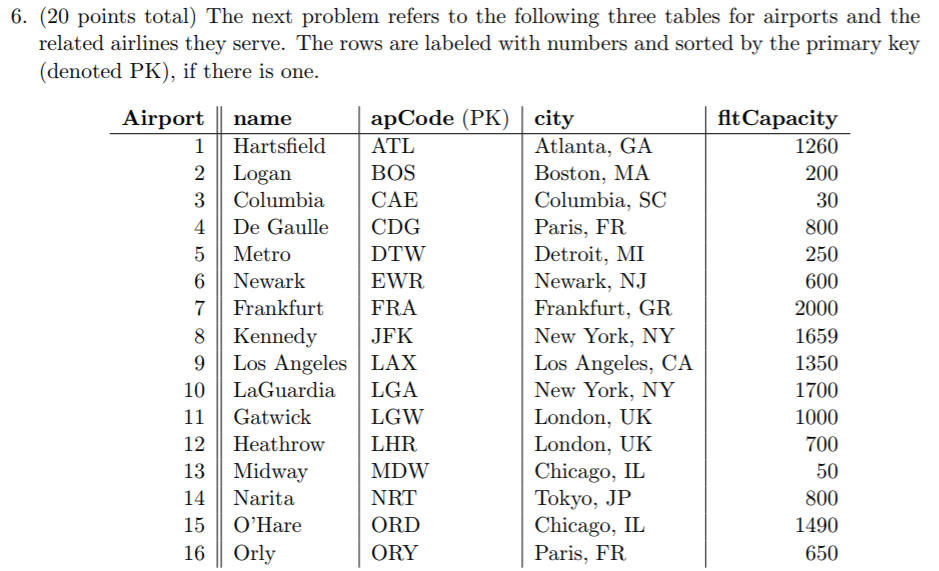
**ADE+ = ADEC**



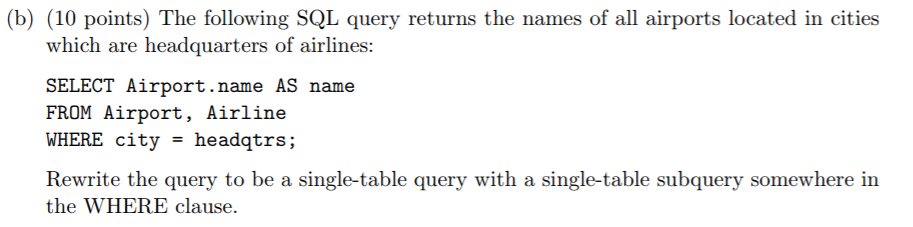


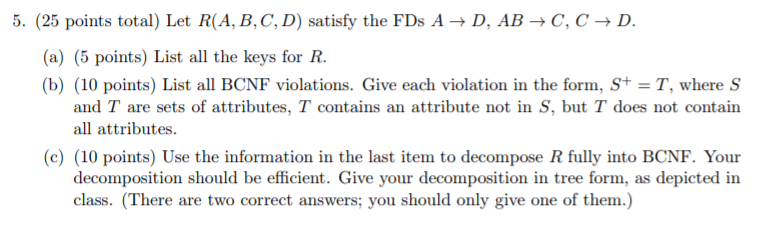












1. **{}+ = {}**

**A+ = AD**

**B+ = B**

**C+ = CD**

**D+ = D**

**AB+ = ABCD**

**AC+ = ACD**

**AD+ = AD**

**BC+ = BCD**

**BD+ = BD**

**CD+ = CD**

**ABC+ = ABCD**

**ABD+ = ABDC**

**ACD+ = ACD**

**BCD+ = BCD**

**ABCD+ = ABCD**

1. **BCNF Violations:**

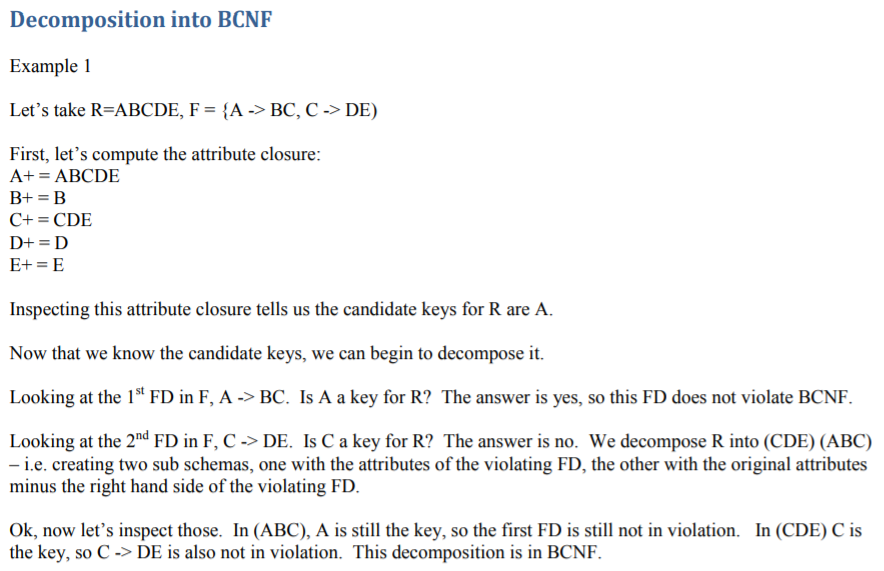
**A+ = AD**

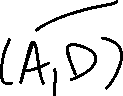
**C+ = CD**

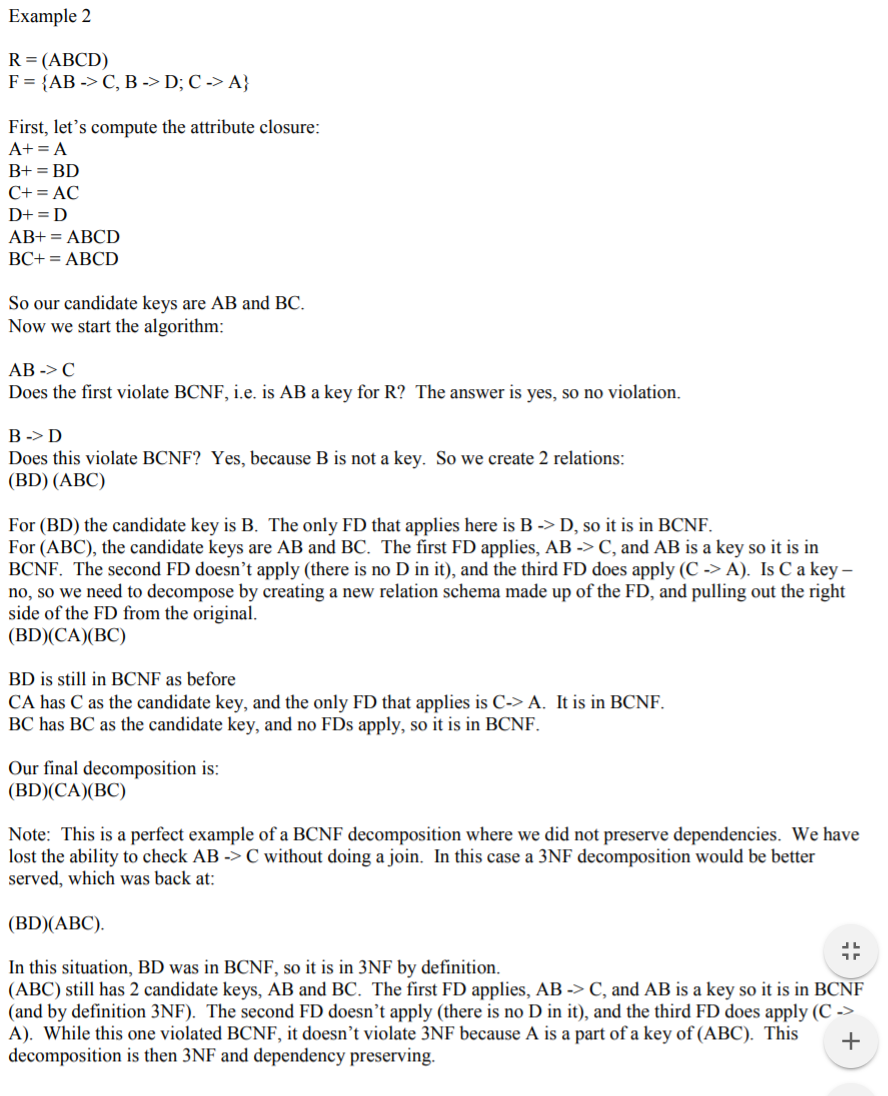
**AC+ = ACD**

**BC+ = BCD**



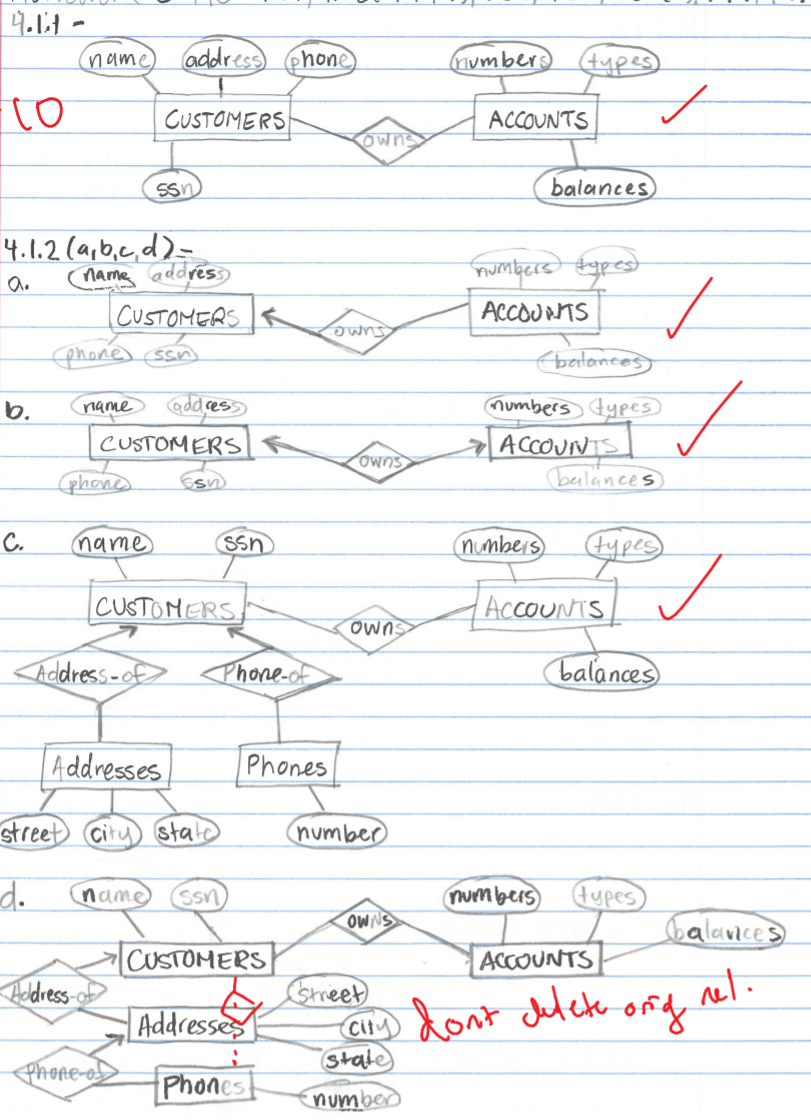


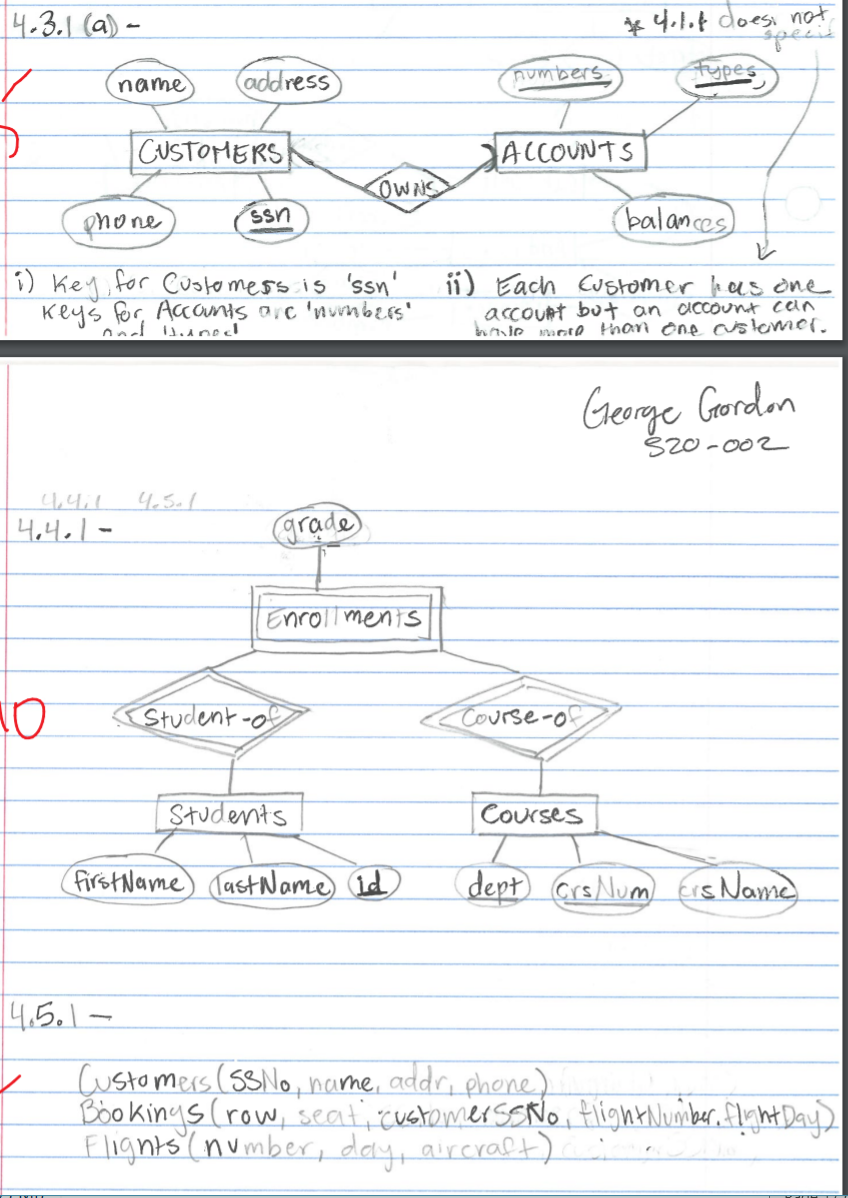




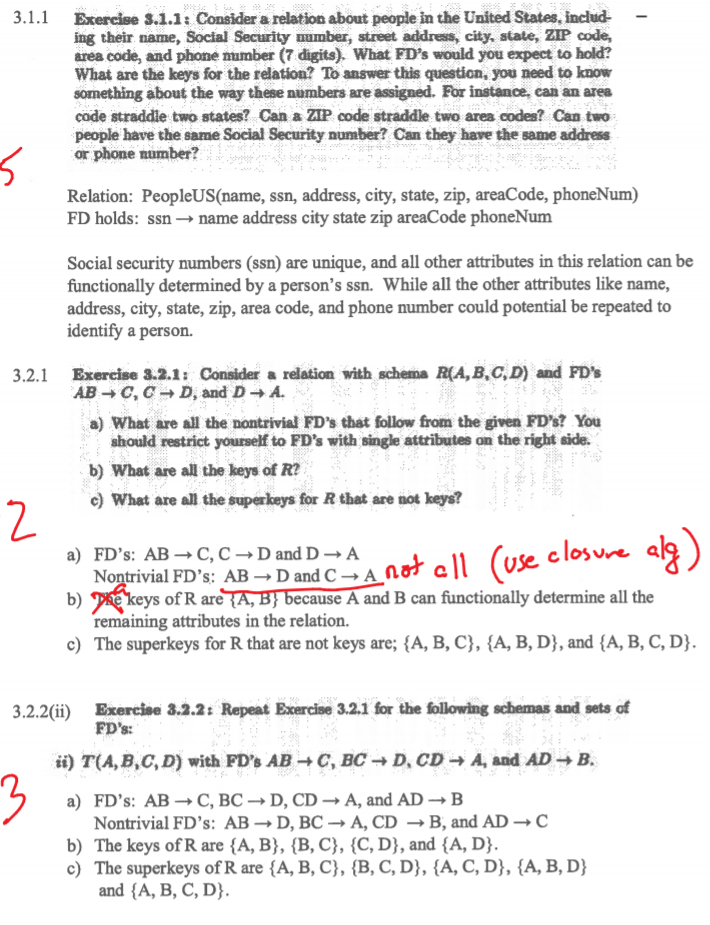
**HOMEWORK 5**

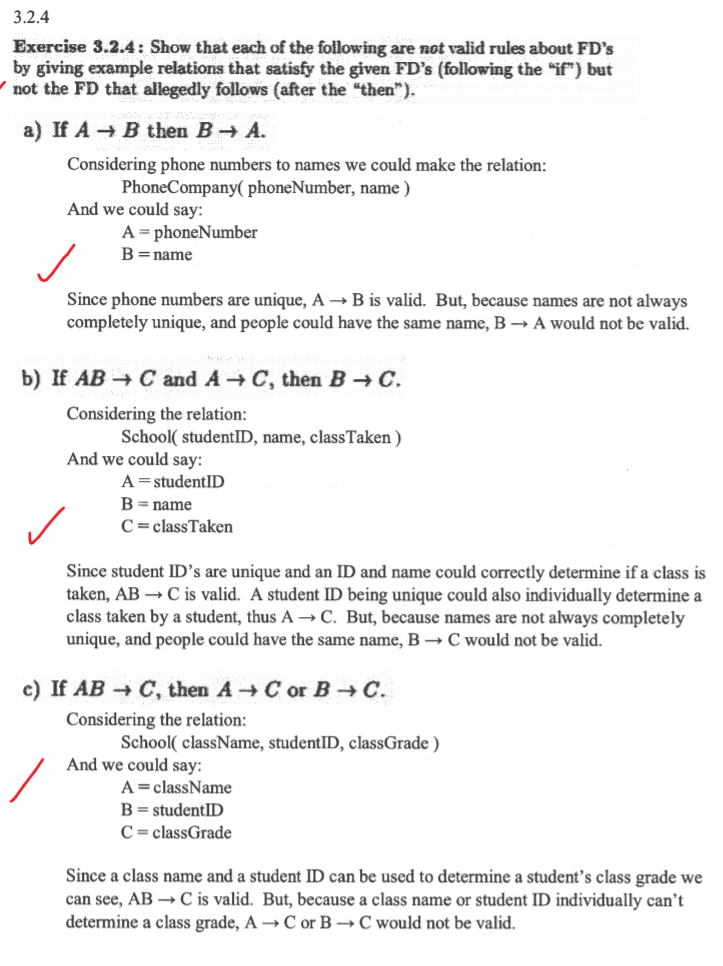


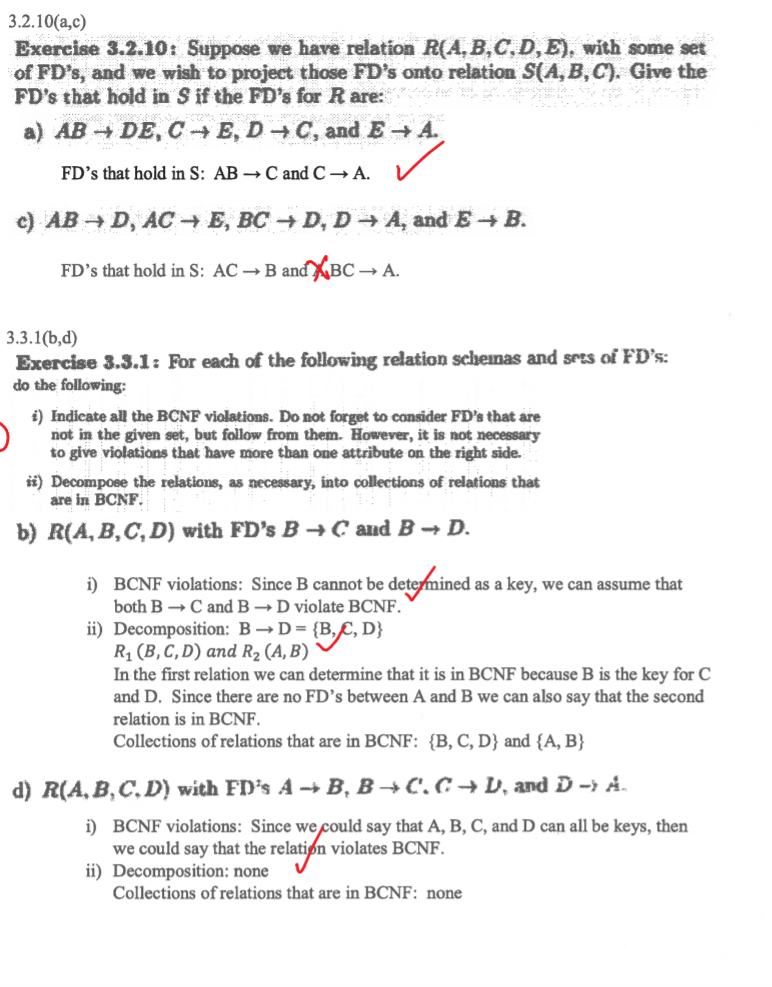


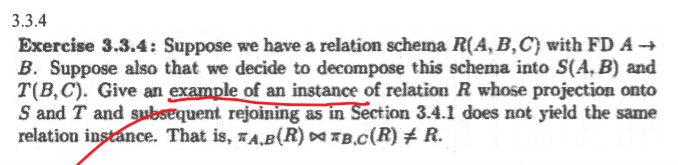


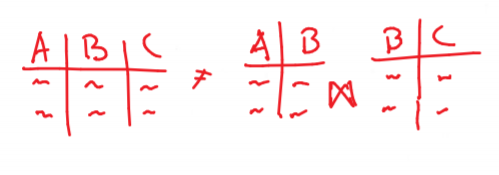
**HOMEWORK 4**



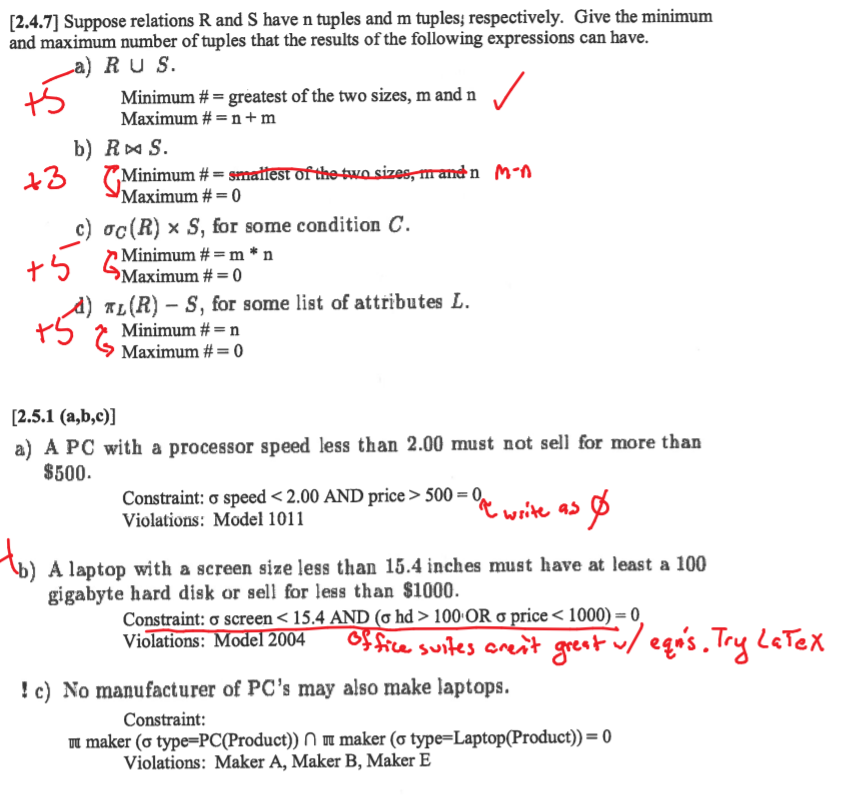








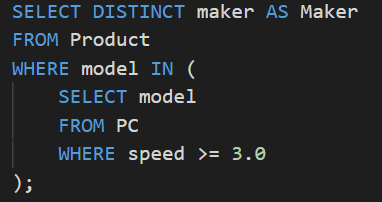
**HOMEWORK 3**



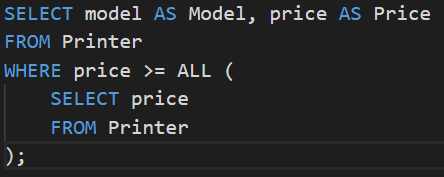
**SQL**

**HW 5 – Exercise 6.3.1**

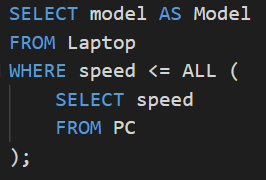
1. **Find the makers of PC’s with a speed of at least 3.0.**



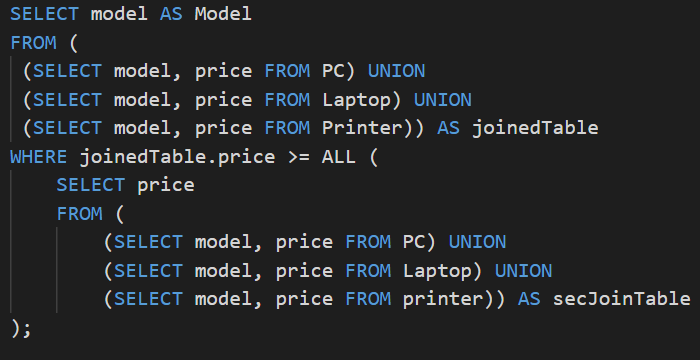
1. **Find the printers with the highest price.**



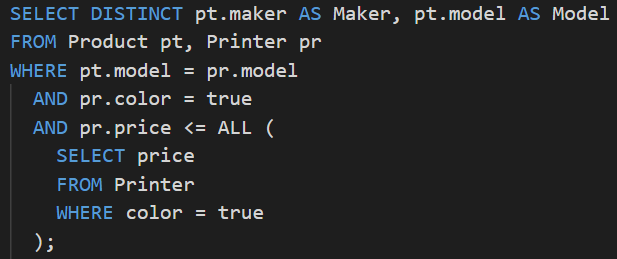
1. **Find the laptops whose speed is slower than that of any PC**



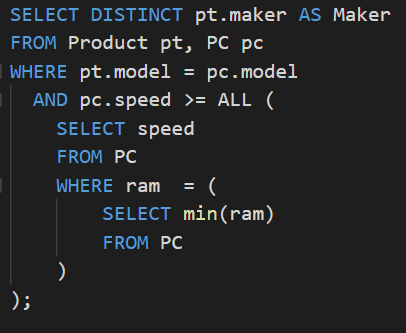
1. **Find the model number of the item (PC, laptop, printer) with the highest price.**



1. **Find the maker of the color printer with the lowest price.**



1. **Find the maker(s) of the PC(s) with the fastest processor among all those PC’s that have the smallest amount of RAM.**



**HOMEWORK 4 – SQL**

