

```

1  Brandon Lasher
2  CSIT565
3
4  HW 2: Relational Design Theory
5  =====
6  ***** Assigned: Sept. 1 *****
7  ***** Due: before class, Sept. 8 *****
8  ***** Submit a soft copy to the Blackboard. Please show the decomposition process for
   better understanding your solutions.*****
9
10 1. Design by Decomposition
11
12
13 Consider the following relational schema:
14
15 Sale(salesman, store, commission_rate, city, date_sold, product, style, color)  // a
   salesman sold a product on a particular day
16 Product(product, style, color, price)  // prices, available styles and colors for
   products
17
18 Make the following assumptions, and only these assumptions, about the real world being
   modeled:
19
20 -- Each salesman works in one store and has one fixed commission rate.
21 -- Each store is in one city.
22 -- A given product always has the same price, regardless of style or color.
23 -- Each product is available in one or more styles and one or more colors, and each
   product is available in all combinations of styles and colors for that product.
24
25 Sale does not contain duplicates: If a salesman sells more than one of a given product
   in a given style and color on a given day, still only one tuple appears in relation Sale
   to record that fact.
26
27 (a) Specify a set of completely nontrivial functional dependencies for relations Sale
   and Product that encodes the assumptions described above and no additional assumptions.
28
29 Sale FDs:
30 salesman->store, commission_rate
31 store->city
32 product->>style,color
33
34 Product FDs:
35 product->price
36 product->>style,color
37
38
39 (b) Based on your functional dependencies in part (a), specify all minimal keys for
   relations Sale and Product.
40
41 These do not assume multi-relations when dealing with style and color, they will be
   added in later questions?
42
43 Sale minimal-key: ( salesman, date_sold, product )
44 Product minimal-key: ( product )
45
46
47 (c) Is the schema in Boyce-Codd Normal Form (BCNF) according to your answers to (a) and
   (b)? If not, give a decomposition into BCNF.
48
49 Sale(salesman, store, commission_rate, city, date_sold, product, style, color)
50 Sale Key: { salesman, date_sold, product }
51 { salesman, date_sold, product, style, color}+ = ( salesman, store, commission_rate,
   city, date_sold, product, style, color )
52
53 salesman->store, commission_rate :: not in BCNF
54 {salesman}+ = store, commission_rate, city
55   S0 = (salesman, store, commission_rate, city)
56   -> With key = salesman
57   FD's are ::

```

```

58         salesman -> store, commission_rate  which is      (salesman,
        store,commission_rate)  This is BCNF since key is salesman and it matches
        our FD
59         store -> city                                This is not BCNF
60
61         -> {store}+ = store, city
62         With key store::
63         FD's are ::
64         store -> city  which is  ( store, city )  This is BCNF as it
        matches our FD
65
66
67
68
69
70
71         S0-{salesman}+ = date_sold, product, style, color
72         S1 = (salesman, date_sold, product, style, color )
73         -> With key = salesman, product
74         FD's are ::
75         product->>style,color  which is (product, style, color ) This is BCNF
76
77         S1 - {salesman, product}+ = date_sold
78
79
80     For Sale(salesman, store, commission_rate, city, date_sold, product, style, color)
81
82     R0 = (salesman, store, commission_rate)
83     R1 = (store, city)
84     R2 = (product, color, style)
85     R3 = (date_sold)
86
87     For Product(product, style, color, price)
88     Product Key: ( product )
89
90     {product}+ = price, style, color , so this is BCNF
91
92     R0 = (product, price, style, color)
93
94
95     (d) Now consider your decomposed relations from part (c), or the original relations if
    you did not need to decompose them for part (c). Specify a set of nontrivial multivalued
    dependencies for relations Sale and Product that encodes the assumptions described above
    and no additional assumptions. Do not include multivalued dependencies that also are
    functional dependencies.
96
97     I am not sure how to answer this question... Because of the multiple styles and products
    in order to be in 4NF form we will need to add relations for each of the multivalued
    relations
98
99     so R2 from Sale will become
100    (product, color) and (product, style)
101
102
103    and R0 from Product will become
104    (product, price), (product, color) and (product, style)
105
106
107
108
109    (e) Are the relations you used in part (d) in Fourth Normal Form (4NF) according to your
    answers for (a)-(d)? If not, give a decomposition into 4NF.
110
111    The full decomposition is:
112
113    For Sale(salesman, store, commission_rate, city, date_sold, product, style, color)
114    R0 = (salesman, store, commission_rate)
115    R1 = (store, city)
116    R2 = (product, color)

```

```
117 R3 = (product, style)
118 R4 = (date_sold)
119
120
121 For Product(product, style, color, price)
122 R0 = (product, price)
123 R1 = (product, color)
124 R3 = (product, style)
125
```