Exercises Database Normalization

---

1. Consider the relation $R(A, B, C, D, E)$ together with the functional dependencies.

$$\begin{aligned} D &\to A \\ E &\to A, D \\ C, D &\to A \\ A, E &\to C \\ B &\to A, D, E \end{aligned}$$

Compute a set of canonical (minimal dependencies).

Solution:
Make the right-hand sides singular:

$D \to A$	$C, D \to A$	$B \to A$
$E \to A$	$A, E \to C$	$B \to D$
$E \to D$		$B \to E$

Minimise the left-hand sides:

1. Drop C from left-hand sides $C, D \to A$ since $D \to A$:

$D \to A$	$D \to A$	$B \to A$
$E \to A$	$A, E \to C$	$B \to D$
$E \to D$		$B \to E$

2. Drop A from left-hand sides $A, E \to \ldots$ since $E \to A$:

$D \to A$	$D \to A$	$B \to A$
$E \to A$	$E \to C$	$B \to D$
$E \to D$		$B \to E$

3. Intermediate result is:

$$\begin{array}{lll} D \to A & E \to A & B \to A \\ & E \to C & B \to D \\ & E \to D & B \to E \end{array}$$

After removing implied FDs, the final result is:

$$\begin{array}{lll} D \to A & E \to C & B \to E \\ & E \to D & \end{array}$$

---

2. Transform the relation in task 1 to BCNF using the synthesis algorithm.

Solution:
We already have a set of canonical FDs.

Maximising the right-hand sides yields:

$$D \to A \qquad E \to A, C, D \qquad B \to A, C, D, E$$

The only minimal keys is $\{B\}$.

We start with the schema $R(A, \underline{B}, C, D, E)$ and we split off violating FDs:

1. $R(A, \underline{B}, C, D, E)$
2. The dependency $D \to A$ violates BCNF, we split:
   $R(\underline{B}, C, D, E), R_2(\underline{D}, A)$
3. The dependency $E \to A, C, D$ violates BCNF of R, we split:
   $R(\underline{B}, E), R_2(\underline{D}, A), R_3(\underline{E}, C, D)$

---

3. Transform the relation in task 1 to 3NF using the synthesis algorithm.

Solution:
We already have a set of canonical FDs:

$$\begin{array}{lll} D \to A & E \to C & B \to E \\ & E \to D & \end{array}$$

We create relations by merging FDs with the same left-hand side (we do not maximise right-hand sides):

$$R_1(\underline{D}, A) \qquad R_2(\underline{E}, C, D) \qquad R_3(\underline{B}, E)$$

Check 1: R_3 contains a key of the original relation. OK

Check 2: no relation is contained in another. OK.