elations that satisfy the following

form (3NF) if, whenever a non-
then either *X* is a superkey or *A* is

h non-key attribute is functionally
her attribute.
e look to see if any non-candidate
unctionally dependent on another
functional dependency exists, we
ute from the relation, placing it in
determinant remains in the original
since the undesirable dependency is
t a superkey and `status` is not part
form the set of relations:

`ajor, credits)`

GURE 6.5(B). In fact, we may decide
all, and instead calculate the `status`
, we simply drop the `Stats` relation,
e that uses the value of `credits` and

iple candidate keys. If we had a second
inal relation, we would have, among

ecNo is a superkey for the relation, so
ity number in the `NewStu2` relation,
attributes there, without violating third

m is the original one developed by Codd
a single candidate key, but it was found
ple composite candidate keys. Therefore,
nal form, named for its developers, Boyce
re of such cases.

rmal Form

y stricter than 3NF.

Codd normal form (BCNF) if, whenever a
X → *A* exists, then *X* is a superkey.

Therefore, to check for BCNF, we simply identify all the determinants and verify that they are superkeys. If they are not, we break up the relation by projection until we have a set of relations all in BCNF. For each determinant, we create a separate relation with all the attributes it determines, while preserving the ability to recreate the original relation by joins by keeping the determinant itself in the original relation.

In our earlier examples, we started by making sure our relations were in first normal form, then second, then third. However, using BCNF instead, we can check for that form directly without having to go through the first, second, and third normal forms. Looking back, we see that for our `NewStudent` relation shown in Figure 6.5(A), the determinants are `stuId` and `credits`. Since `credits` is not a superkey, this relation is not BCNF. Performing the projections as we did in the previous section, we made a separate relation for the determinant `credits` and the attribute it determined, `status`. We also kept `credits` in the original relation, so that we could get the original relation back by a join. The resulting relations are BCNF. For the relation `NewClass` shown in Figure 6.4(A), we found the determinants `classNo`, which is not (by itself) a superkey, and `stuId`, also not a superkey. Thus the `NewClass` relation is not BCNF. We therefore created a separate table for each of these determinants and any attribute each determined, namely `Stu` and `Class2`, while also keeping a relation, `Register`, that connects the tables. The relations resulting from the projections are BCNF. If there is only a single candidate key, 3NF and BCNF are identical. The only time we have to be concerned about a difference is when we have multiple composite candidate keys.

Let us consider an example involving candidate keys in which we have 3NF but not BCNF.

```
NewFac (facName, dept, office, rank, dateHired)
```

For this example, shown in FIGURE 6.6(A), we will assume that, although faculty names are not unique, no two faculty members within a single department have the same name. We also assume each faculty member has only one office, identified in `office`. A department may have several faculty offices, and faculty members from the same department may share offices. From these assumptions, we have the following FDs. Again, we are dropping set braces, ignoring trivial FDs, and listing dependents with the same determinant on the right-hand side of the arrow. Recall that when multiple attributes appear on the left-hand side of the arrow, it means the combination is a determinant.

```
office → dept
facName, dept → office, rank, dateHired
facName, office → dept, rank, dateHired
```