

3 Buddy Problem

February 12, 2021

1 The Problem

Everyone needs a coding buddy. So we have a column associated with each user called “Buddy”. But now we need a group to have 3 people. How do we represent this group of three people?

2 Some Solutions

- I. Add a column for the third person
 - A. What happens if we need to put a fourth person? fifth?
 - B. Do we really want so many empty cells associated with each of the other pairs?
 - i. Let’s try and avoid nulls
- II. Comma separate the buddies columns
 - A. Not atomic
 - i. 1keydata.com
 - ii. [dba.stackexchange](https://dba.stackexchange.com)
- III. Create a new row for every pair within the new group
 - A. This is going to get really ugly really quickly. Imagine we need 1 group with 4 people. 8 rows?
 - B. $\frac{n(n-1)}{2}$ pairs for any group of size n.
 - C. $O(n^2)$ space for this task should be viewed as unacceptable.
- IV. Create a new column called “Team_Name”
 - A. Create a new table, “Teams” with appropriate attributes and a primary key
 - B. There will be a one to many relationship between the CLASS_ROASTER table and the TEAMS table via a foreign key relationship.
 - C. We should note that the team does not need to know who belongs to it

3 My Opinion

I believe solution IV to be the correct solution. It will allow us to maintain a scalable, robust, minimal to no redundancies, and reusable design. It is now very easy and hopefully obvious how to add information about the teams going forward.