Coding Therapy for Software Developers

Steven Feuerstein
PL/SQL Evangelist, Quest Software
www.ToadWorld.com/SF (PL/SQL Obsession)
www.PLSQLChallenge.com (daily PL/SQL quiz)

I am not making light of therapy

- **Therapy can be a very important and meaningful process for healing pain and distress.**

- **I am not mocking *real* therapy nor do I suggest that my session today can substitute for a session with a real therapist.**

- **But perhaps I can help you with a few code-related issues....**

Why might therapy help developers?

- **Sure, some software is generated.**
- **But almost all software continues to be written directly by human beings.**
- **Software construction is a wholly intellectual endeavor.**
  - That is, to write software we rely almost exclusively on our brains.
- **Consequently, human physiology and psychology play an enormous role in how software is written - and *mis*-written.**

Coding Therapies for Today

- **Game Therapy**
  - Learning through fun
- **Dream Therapy**
  - Let your subconscious do the "heavy lifting."
- **Don't be afraid to ask for help.**
- **Shock therapy**
  - SQL is *not* your best friend.
- **Couples therapy**
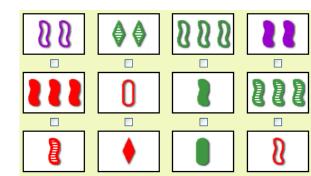 - Developer and Database Administrator
  - Developer and Manager

3

Game Therapy

- **Games can be quite helpful in the learning process.**
  - We like to compete (even if only against ourselves), have fun, win prizes.
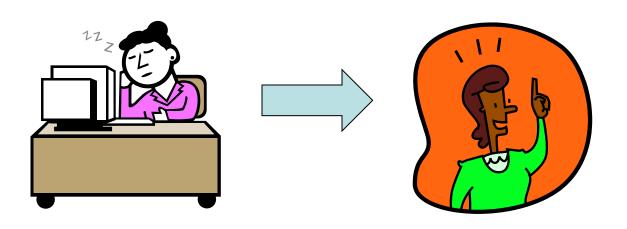
- **Mastermind**
  - Classic of deductive logic
  - Will improve your debugging skills

- **Set (www.setgame.com)**
  - Pattern analysis and resolution
  - Will generally improve your development skills and quality of code.

- **PL/SQL Challenge – PL/SQL, SQL and APEX quizzes!**

4

Dream therapy for developers

- **The worst way to deal with a problem is to worry at it (for hours or days).**
    - Sometimes *conscious* attention only makes it harder to find the solution.

- **Maybe you should sleep on it, instead.**
    - There's a good chance that when you wake up the answer will be waiting in your conscious "in box."

Change the Body, Change the Mind

- **Change what your body does, and you change how your mind works.**

- **If you can't go to bed and you're stuck on a problem, then...**
  - Take a walk, run around the block, do some stretches or sit-ups, *anything* but stare at your screen.

- **Best of all: leave the sterile environment of computers and get out into the world.**
  - We spend almost all of our time completely, physically disconnected from our own planet.

6

Don't be afraid to ask for help.

- **Our evolved survival instinct urges us to hide weakness.**

- **On top of that, we software developers are supposed to be really smart.**
  - We are the wizards of modern society.

- **Unfortunately, ignorance leads directly to bugs and sub-optimal code.**

It's OK to say "I don't know. Help me!"

- **Just *thinking about* asking for help will often do the trick.**
- **Most people *like* to be asked to help.**
  - It makes them feel *valued*.
  - It strengthens the team as a whole.
- **It may not really matter *who* you ask for help.**
  - If there are no programmers handy, ask your spouse or parent or child to be a sounding board.
  - Or write an email. By the time you finish writing it, you will likely have found the answer to your problem.
  - The important thing is to get the issue *out of your head*.

Follow the Thirty Minute Rule

- **We are usually too deeply *inside* (and part of) the problem to step back and take a fresh look.**

- **If you can't fix a bug in 30 minutes, ask for help.**
  - For "trivial" bugs, "give up" after just a few minutes!

- **Senior developers and managers must take the lead.**
  - Ask more junior members for help. Show that you are fallible, that you can learn from anyone and everyone.

Shock Therapy

- **Sometimes you need to be given a shock to be able to break out of your ruts, and gain a new perspective on your life.**
  - Same with software.
- **So here's your 1st shock therapy of the day:**

What? No SQL for PL/SQL developers?

- **More or less.**

- **You should avoid writing SQL statements all over your application.**

- **Every SQL statement you write is a *hard-coding*.**
  - And a more dangerous, problematic hard-coding than hard-coded *literals*.

- **You shouldn't repeat "170546" and you shouldn't repeat "SELECT * FROM...."**

- **Yes, I know: shocking!**

11

What's a PL/SQL developer to do?

- **Create a separate data access layer.**
    - Also known as "table API" or "transaction API".

- **Instead of writing SQL statements, call functions and procedures that _do_ the work for you (services)....**
    - Hiding your implementation (it will change)
    - Strengthening your error handling (you only write it – or generate it – once)
    - Increasing your productivity.

- **Look for ways to generate as much of this code as you can.**

Download Quest CodeGen Utility from PL/SQL Obsession

12

More shocks to the system...

- **You should consider every hard-coded VARCHAR2(N) declaration to be a *bug*.**
  - Unless it's the "single point of definition."

```
DECLARE
    l_full_name VARCHAR2(100) :=
        first_name_in || ' ' || last_name_in;
    l_big_string VARCHAR2(32767);
```

Three bugs!

- **What should you do instead?**
  - Use %TYPE, %ROWTYPE and SUBTYPEs.

```
DECLARE
    l_full_name employees_rp.full_name_t :=
        employees_rp.full_name (first_name_in,
                                last_name_in);
    l_big_string plsql_limits.maxvarchar2_t;
```

plsql_limits.pks
explimpl.pkg

13

Couples Therapy: Developer and DBA

- **Can't we all just get along?**
- **Apparently not....**
  - Developer: "Why can't I just [add that column | drop that table | mess things up] whenever I want?"
  - DBA: "First, tune your code."
- **Stay on good terms with your DBA.**
  - Let's face it; a DBA has lots more responsibility and privileges (which translate to "power" in a database).
  - You can't win a battle with a DBA.
- **Application development should be a *collaborative* process.**

Developer and DBA: a close collaboration is best

- **As you change the way you write your code, you should notify your DBA.**
    - Start to work with collections? Notify!
    - Using FORALL and BULK COLLECT? Notify!
    - Discovered Oracle11g function result cache? Notify!
- **Ask (negotiate) for the resources you need to analyze code impact yourself.**
    - For example, with access to just a handful of V$ views, you can monitor PGA memory utilization of your PL/SQL code.

plsql_memory.pkg
grantv$.sql
11g_frc*.*

15

Couples Therapy: Developer and Manager

- **Are you satisfied with the way your manager communicates with you?**

- **Which of these does your manager ask:**

How *maintainable* is your code?

When will you be *done*?

How can *I help you* get your job done?

How *thoroughly* have you tested your code?

Can you make it go *faster*?

Did you follow the coding *standards*?

16

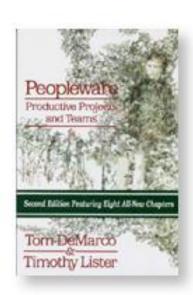# Developer and Manager: Pretense is Critical

- **Seems like some managers need to know that you, the developer, know "who's boss."**
    - Even though, really, the only point of a manager is to help *you* succeed (after all, who does the *real* work?).
- **So avoid *direct challenges* to authority and instead...**

"Thank You For Being a Great Manager!"

- **Buy your manager a copy of Peopleware as a thank you gift.**
  - A classic text for building development teams
- **Ask frequently for help, advice and direction.**
  - "How can I improve my testing?"
- **Ask for coding standards and mechanisms to verify that you are following them.**
  - Standards *improve* your quality of programming life.

Coding Therapy:
You can't escape your own brain!

- **Our brains' physiology and psychology have an unavoidable and *large* impact on the way we write code.**

- **Don't fight it; instead, understand it, accept it, adapt to it.**

19