# Web Development Best Practices

(or “We're Not Perfect, But It's Worth A Shot”)

# Be Kind, Rewind

- Be Consistent (spacing, formatting)
- Make it Maintainable (create and use functions)
- Know Your Tools (use appropriate constructs)
- Care (write code of which you can be proud)
- Comment (explain complicated sections, srsly!)
- Self-Document (items describe themselves!)
- **Use** Version Control (explain why it changed)

# Unbroken Eggs in a Basket

- Avoid Code Duplication (fixing one version of the function will leave all other copies broken)
  - Replacing repeated code with a function can greatly improve code legibility / simplify QA
- Validate Input
- Check Permissions
- Escape SQL Input / Use Prepared Statements
- Escape User Web Content - htmlspecialchars()

# Consistency

- Always indent the same amount (and with the same characters)
- When editing, use the style of the application
  - Follow the coding style (esp. for external apps)
  - Use existing database connections if possible
  - DO NOT shoehorn yet another code library into an application which can solve the problem
    - Unless you can convince the owner to switch the whole application, use existing infrastructure

# Version Control (Subversion)

- Easy to manage and deploy changes; much better than CVS
  - svn checkout https://repo.example.com/svn/test
  - svn status
  - svn add
  - svn diff
  - svn commit
    - Leave a comment about what changed! At least touch on every notable change.

# Databases (e.g. MySQL)

- Evaluate queries! For SELECTs, prepending EXPLAIN will tell you that database's plan.
- Use an index! Particularly for large tables or joins, an index is a life saver.
- Try to GROUP/ORDER BY with an index. Remember that index field order matters.
- Reduce the number of queries! If you are doing a query in a loop, you're probably wasting a lot of time. If possible, combine like queries.

# PHP Version

- PHP4 is Dead (avoid PHP4 libraries)
  - PEAR::DB / PEAR::MDB / PEAR::MDB2
  - PEAR::SOAP
  - PEAR::XML_RPC / PEAR::XML_RPC2
- PHP5 (there's an extension for that!)
  - PDO
  - SOAP
  - XML-RPC

# PHP Functions

- array_push() is bad! (use $myarray[] = $item;)
- split() is bad! (use explode, str_split, preg_split)
- array_key_exists()? (you probably want isset())
- isset(), empty() are good! (language constructs)
- for ($i = 0; $i < count($rows); $i++) {}
  - foreach ($rows as $row) {}
    - OR (if using valuable keys)
  - foreach ($rows as $key => $row) {}