

# Thou shalt NOT

## !Test

thus sayeth,  
~winter-paches

<https://www.ngzax.com/>





# History of Testing

- Pre-History
- Dawn of History
- Renaissance
- Today



# A small demo of Dynamic Unit Testing



# Reasons to Test... *USING CODE*

- Prevent Cruelty
- You can't *Progress* unless you *Regress*... Slow down to speed up
- You definitely don't want to write *Documentation*, do you?
- Stop dreading merge conflicts and Integration



# Reasons to Test... FIRST

- It's all about the Interface.
- You can refactor
- He who tests last, eventually doesn't test.



# What you will get

- Improved Confidence
- Faster Feedback
- Better Sleep



## Other Stupid Test Tricks...

- Have tests that document "broken" APIs and features that start failing when the upstream is fixed.
- Have "canary" tests to alert you when APIs have changed.
- Make sure you are handling all logic cases
- Make sure you don't have worthless code.



# General Tips & Tricks

- Keep Tests Small
- Keep Tests Fast
- Keep Tests Independent
- Use Descriptive Names
- Make sure it fails!



# %Gall App / Hoon Testing Tips

- Prefer `expect-eq` to `expect`
- Implement `++ as-tape` in your Cores
- Use a Door and a fake bowl.
- Create a `test-data` structure for `++ setup` and `++ expects`
- Test poking your Agent with fake state and moves.
- Test scrying your Agent with fake state, moves, and calling `on-peek`.
- Test Boundary Conditions... see Jack's USTJ article.



# References

- "Writing Robust Hoon — A Guide To Urbit Unit Testing by Neal Davis"  
<https://medium.com/dcspark/writing-robust-hoon-a-guide-to-urbit-unit-testing-82b2631fe20a>
- "A Philosophy of (Inductive) Testing by Jack Fox"  
<https://urbitsystems.tech/article/v01-i01/a-philosophy-of-inductive-testing>
- "The Costs and Benefits of a Unit Testing Culture"  
<https://martinfowler.com/articles/testing-culture.html#costs>
- "POKA YOKE - Applying Mistake Proofing to Software"  
<https://techie-notebook.blogspot.com/2012/07/poka-yoke-applying-mistake-proofing-to.html>

## Hoon Test Examples

re.hoon by ~macrep-racdec

<https://github.com/lynko/re.hoon/blob/master/src/tests/lib/regex.hoon>

test.agent by ~palfun-foslup

<https://github.com/tloncorp/tlon-apps/blob/develop/desk/lib/test-agent.hoon>

%rank by ~winter-paches

<https://github.com/Zaxonomy/rank/tree/master/src/tests>