Peer Instructor Training

January 30, 2020

Meet Your Facilitators

Daniel Kerchner

Senior Software Developer

Kevin Knudsen

Director of Academic Commons

Laura Wrubel

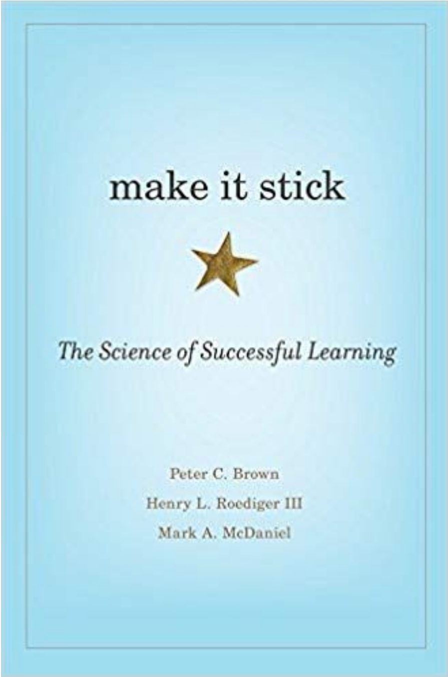
Software Development Librarian

Goals for this Workshop

1. Identify and describe behaviors to create a positive learning environment.
2. Identify helpful and unhelpful behaviors in teaching learners to code.
3. Identify and avoid demotivating behaviors.
4. Model inclusive language and helpful behaviors in teaching code or software.
5. Be able to give feedback about others' teaching and receive feedback on your teaching.

Learning How to Learn

Learning is an acquired skill, and the most effective strategies are often counterintuitive



make it stick

The Science of Successful Learning

Peter C. Brown

Henry L. Roediger III

Mark A. McDaniel

Learning is Misunderstood

- Learning is deeper and more durable when it is effortful.
- Massed practice is among the least effective ways to learn a new skill.
- Trying to solve a problem before being taught the solution leads to better learning.
- All new learning requires a foundation of prior knowledge.

Brown,
Roediger, &
McDaniel, 2014

Different Types of Learners

Experts

- Notice meaningful patterns of information.
- Well-organized mental models.
- Knowledge is "conditionalized."

Novices

- Surface-level understanding of processes.
- No or disorganized mental models.
- Ideas are not or have weak connections.

Think back to a time where a professor/instructor **decreased** your motivation. Discuss the situation with a partner.

Motivation

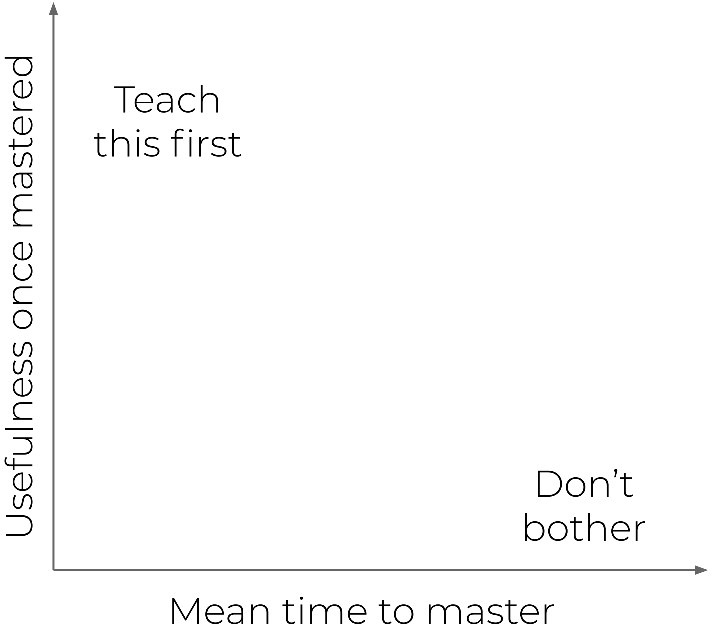
Creating a Positive Learning Environment

- Presenting the instructor as a learner.
- Establishing norms for interaction
- Encouraging learners to learn from each other.
- Acknowledging when learners are confused.

The Carpentries, 2018-2020

The Carpentries, 2018-2020

Teach the Most Useful First



Motivation in Coding Instruction

Think back to a computational or other coding course you took in the past. What is one thing the instructor did that motivated you?

Motivation in Coding Workshops

DO

- Relate your examples and content to things useful in real life.
- Give lots of time for tasks, allow time for failure.
- Show your own enthusiasm for the topic.
- Share your own mistakes.

DON'T

- Say negative or scornful things about a particular OS or software. (e.g. Windows vs. Macs).
- Take over a learner's keyboard.
- Use terms like "easy", "simple", and "just".

The Carpentries, 2018-2020

Dismissive Language

- You are not your learners. What is easy to you is not simple to them.
- Using the word “just” in explanations signals to learners you think their problem is trivial and that they therefore must be stupid.
- Ask "What questions do people have?” rather than “Does anyone have any questions?” Having questions is normal and expected.

The Carpentries, 2018-2020

Live Coding
Is a SKILL!

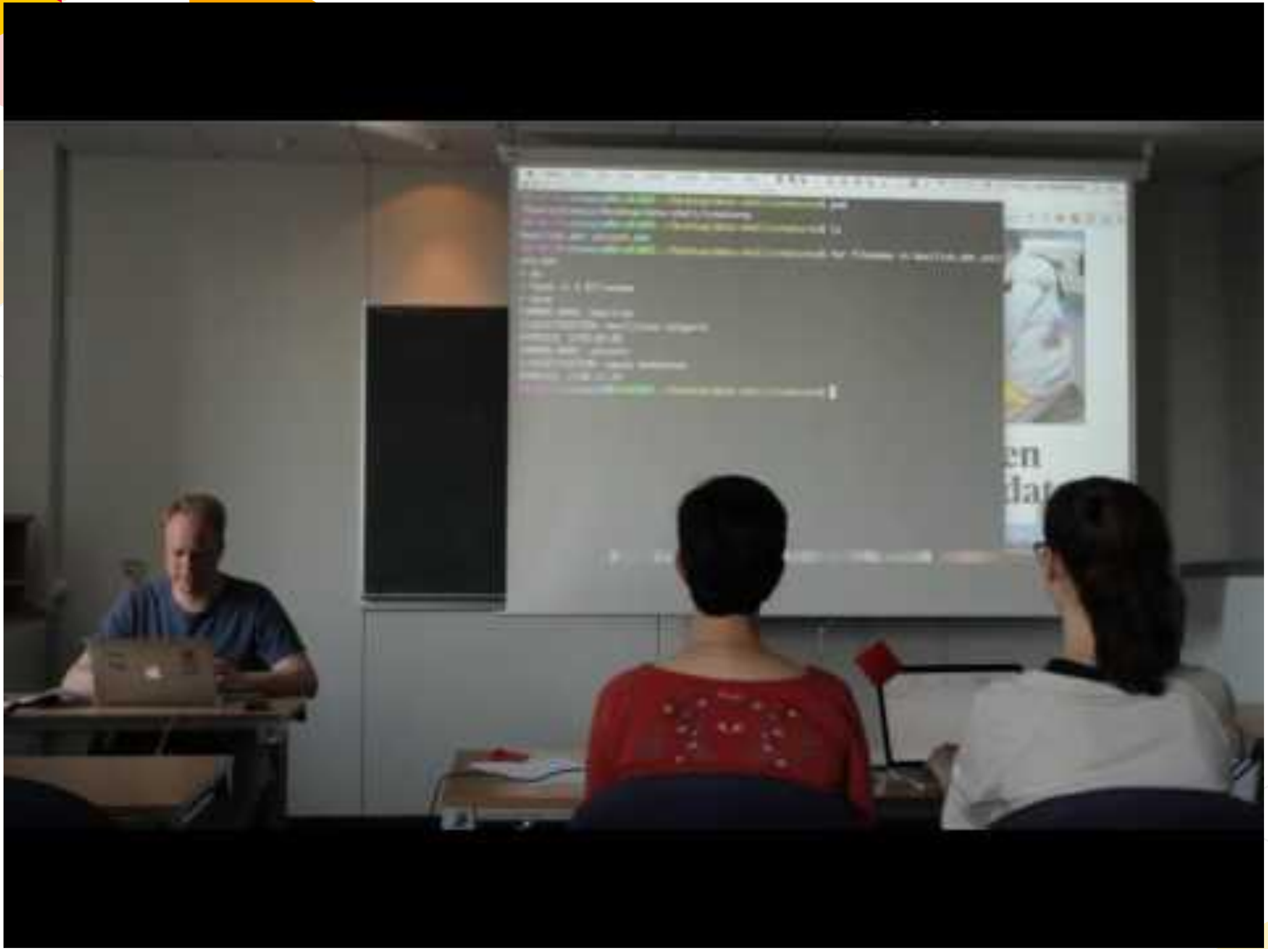
Live Coding vs. Slides

+ More compelling
+ Responsive
+ Tacit knowledge
+ Pacing
+ Mistakes

Challenges:

- Learners have to listen + type
- Instructor must:
  - Improvise
  - Think aloud
-

The
Carpentries,
2018-2020



TOP 10 TIPS
for participatory live coding!

TOP 10 TIPS for Participatory Live Coding

#1: Stand up! Use a mic!

- It's more interactive
- It's less boring

The Carpentries, 2018-2020

TOP 10 TIPS for Participatory Live Coding

#2: Speak what you type

- This slows you down
- Allows everyone to copy you
- Don't copy/paste code

The Carpentries, 2018-2020

TOP 10 TIPS for Participatory Live Coding

#3: Match your learners' environment

- Use default appearance where possible
- This avoids cognitive load
- Avoid shortcuts

The Carpentries, 2018-2020

TOP 10 TIPS for Participatory Live Coding

#4: Use your screen wisely

- BIG FONT
- Black text, white background is easier to see
- Use all screen space
- Beware of working too low down
- Minimize glare in the room

The
Carpentries,
2018-2020

TOP 10 TIPS for Participatory Live Coding

#5: Use Diagrams

- OK to draw on a board
- Helps organize concepts

The Carpentries, 2018-2020

TOP 10 TIPS for Participatory Live Coding

#6: Avoid distractions

- Clean up laptop desktop
- Turn off notifications (both laptop and phone)
- If you have a script, print yourself a paper copy

The Carpentries, 2018-2020

TOP 10 TIPS for Participatory Live Coding

#7: Stick to the material

- Otherwise you'll:
  - lose the value of well-planned material
  - run into the unexpected
  - risk your schedule

TOP 10 TIPS for Participatory Live Coding

#8: Look for (and help) stragglers

- Use sticky notes to surface learners who are falling behind

The Carpentries, 2018-2020

TOP 10 TIPS for Participatory Live Coding

#9: Embrace errors / mistakes

- Errors are opportunities to learn
  - more about coding, but also about
  - troubleshooting and
  - perseverance
- Use positive error framing
- Demonstrates that you are a learner
- Teach the process of troubleshooting!

The Carpentries, 2018-2020

What if I make an error?

Errors are:

- Teaching opportunities (coding + troubleshooting)
- *Integral* to learning
- Opportunities to demonstrate perseverance and growth mindset

Positive error framing encourages learners to see errors as a natural part of learning!

The Carpentries, 2018-2020

Errors/mistakes are rich opportunities

Errors allow you to show:

- Not being intimidated by a "cryptic" error message - read it slowly a few times
- The process you go through to troubleshoot / problem solve

The Carpentries, 2018-2020

TOP 10 TIPS for Participatory Live Coding

#10: Have fun

- Ok to improvise (within reason - see #7)
- Positive language (even humor)
- The more you know your material, the more relaxed you will be

Let's evaluate some live coding

2x2 rubric

Let's PRACTICE some live coding!

Find a partner and a room to practice in

1. Spend 5 minutes preparing to teach the **2-minute concept** you chose earlier.
2. Each person **teaches their segment** to the partner, as if it were a class. (Your partner can observe your laptop if you are live coding.)
3. **Set a timer** and keep a strict 2 minute time-limit.
4. The person not teaching should **use the feedback matrix to record feedback**.
5. After each person has presented, **give each other feedback** on content and presentation.

Resources

- Carpentries Instructor Training curriculum
- The Science of Learning report
- Advice on developing workshop content: consult with today's workshop instructors plus other workshop instructors:

Dan Kerchner

Kevin Knudsen

Laura Wrubel

Ann James

Kean McDermott

Megan Potterbusch

Dolsy Smith

Wrap-up

- **Green sticky:** what went well?
- **Pink sticky:** what should we change for next time?