Welcome to Bootcamp Prep!

Fullstack Academy

Course Overview: Topics

#	Topic
1	Introduction, Tidy Code
2	Loops, Debugging
3	Coercion and Truthiness
4	Scope
5	Arrays I
6	Arrays II
7	Objects

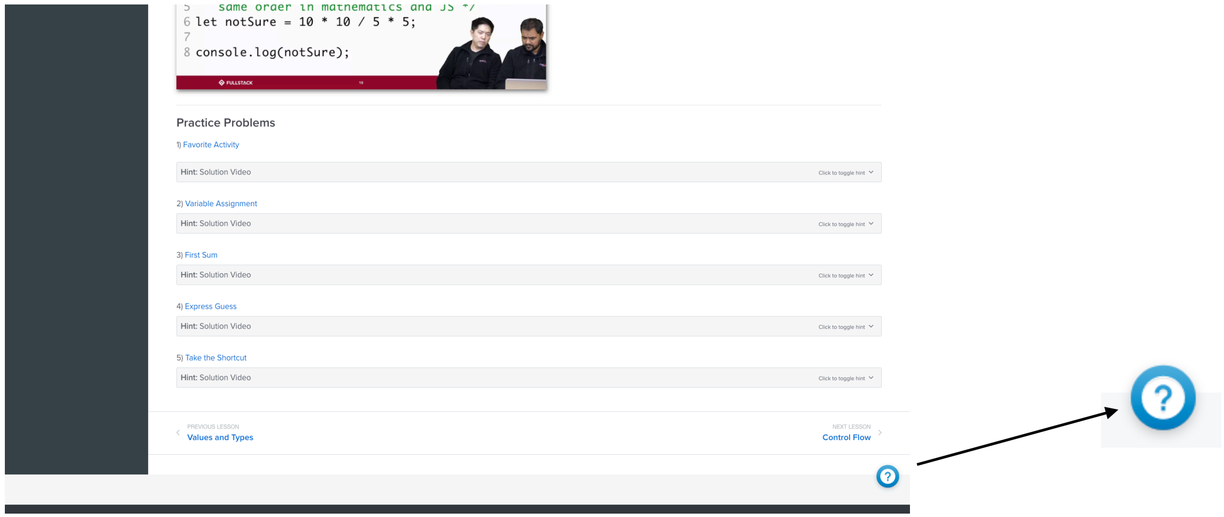
#	Topic
8	Object Methods
9	Passed-by-value Passed-by-reference
10	Higher-order Functions I
11	Higher-order Functions II
12	Recursion I
13	Recursion II
14	Next Steps, Happy Hour Interviews

Course Overview: Format

1. Watch the pre-recorded lecture (if time permits)
2. Attend the interactive lecture (required)
3. Work on the workshop with your partner (required)
4. Watch the workshop review videos and compare your answers with the provided solutions (strongly recommended)

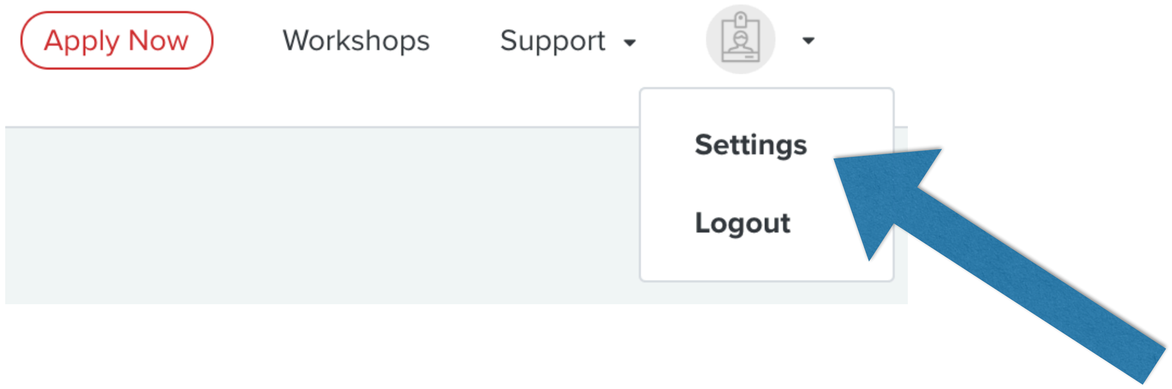
Workshops

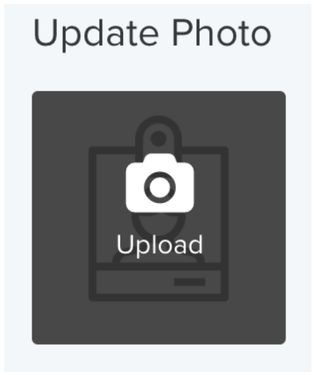
- Workshops are not intended to be finished
- You will be randomly paired with a different partner or small team for each workshop
- Instructors and fellows available via the Help Desk



Workshops: LearnDot Picture

- Add your picture to LearnDot to help instructors find you during help tickets!





Workshops: pair programming

- **Pair programming**
  - 2 programmers, sharing the same computer, work collaboratively on all aspects of software development (Williams and Kessler 2000).
  - "Driver" writes code
  - "Navigator" plans code and reviews it as it is written
  - Switch every 10 minutes, or every time you finish a workshop problem.
- **At Fullstack, you switch partners for each workshop**

Workshops: pair programming

- **Why use pair programming at Bootcamp Prep? Shouldn't I practice on my own?**
  - "Teach to learn" is a central aspect of Fullstack's instructional philosophy
  - Communicating about code is just as, if not more important, than actually writing it
  - Solving problems with a partner is the best way to prepare for an admissions interview at a top bootcamp
  - Most top bootcamps, and a growing number of employers, use pair programming every day

Ice Breaker

- Say your name.
- Say where you are from
- Why you're here.
- An interesting fact about yourself (if you want)

Course Overview: Projects

- Two projects will be introduced as we progress through the course
- 100% optional
- Good opportunity to use what you're learning to build larger programs

Course Overview: Practice Assessments

- It's meant to help you prepare for the real admissions assessment
- A risk free assessment will also be available to you on LearnDot after you finish Bootcamp Prep

Fullstack Community Values

- Be patient. With yourself and others.
- Ask questions (even “dumb” ones)
- Trust the process. Be on time.
- Help others. Teach others.
- We like to have fun, and…

Fullstack Community Values

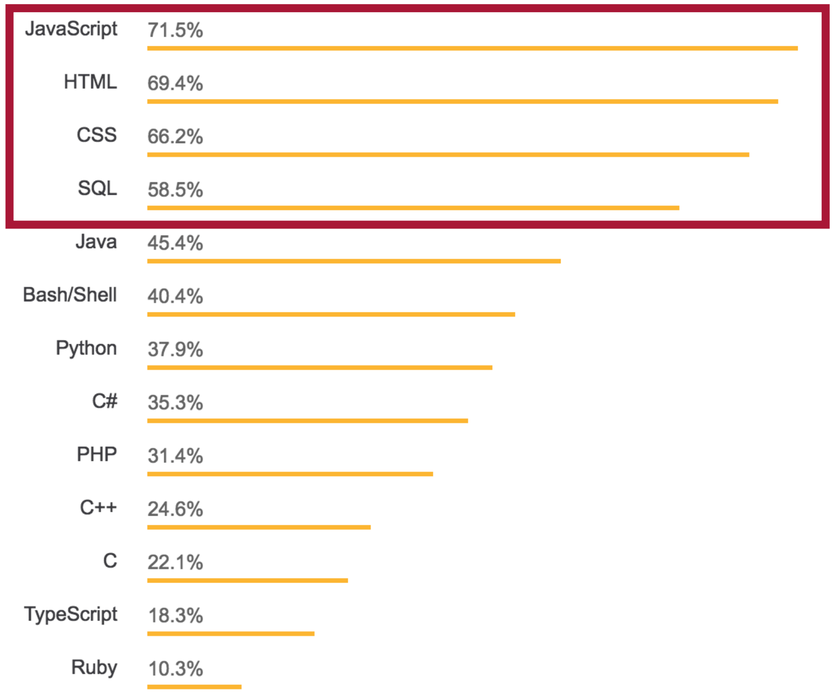
- Be mindful of:
- Subtle “isms”
- E.g. Sexism, Racism, Ageism
- Be Professional - “No Asshole Policy”
- You are each other’s most valuable resource
- No NSFW content. When in doubt, leave it out.

Why Javascript?

Most Popular Technologies

Programming, Scripting, and Markup Languages

All Respondents | Professional Developers



JS === most popular language in the world!

FSA/GHA curriculum

Ruby is great for beginners but much less widely used

Stack Overflow 2018 Developer Survey Results

Why is JavaScript so Popular?

- Have you used this thing called the internet?
- Powers the majority of web applications
- Atwoods Law: "Anything that can be written in JavaScript, will eventually be written in JavaScript."
- Runs almost anywhere on the “full stack”: web browsers and servers
- The first 7/10's of JavaScript is relatively learner friendly

ES5? ESNext?

- **ES5 and ESNext are different versions of JavaScript**
  - **function vs () => {}**
- **JavaScript versions are backwards compatible**
  - Older features work in newer environments
  - Newer features may not work in older environments
- **Fullstack teaches "ESNext"**
  - Immersive programs continuously updated to cover the latest JS features
  - BCP program covers some newer features too, but only those that are helpful for beginning developers — don't expect in-depth or comprehensive coverage of new JS features in this course

Getting the most from this course

- Complete the BCP Pre-work curriculum first!
- Do the pre-readings and watch the pre-recorded lectures
- Enjoy the "productive struggle" - This is gonna be hard, try to enjoy the fact that you're facing a difficult task that can completely alter your life in a really big way.
- Ask for help when you need it (otherwise, why are you here?)
- Don’t look at solutions until you give problems an honest try
- Go back to unfinished workshop problems and finish them when time permits
- Watch the solution videos, even for problems you think you got right
- Review the solution code

Tidy Code Tips

Why write tidy code?

- **Easier and faster to debug and read your code**
  - Engineers spend a lot more time reading and debugging code than actually writing it!
- **Demonstrates qualities attractive to bootcamps and employers:**
  - Professionalism: no longer a "beginner"
  - Attention to detail
  - Pride in work

```
this function is amazing!
```

Indentation

```
1  // indent one level inside every code block
2  // code blocks start with a { and end with a }
3
4  function amazingFunction() {
5    // inside the function block!
6    if (true) {
7      // inside the if block!
8      console.log('this function is amazing');
9    }
10   // outside of the if block
11 }
12
13 // outside of the function block
14 amazingFunction();
```

Variable Naming

1
2
3

```
// use camelCase when defining variable names
let myFavoritePlace = 'Fullstack';

// don't use ambiguous variable names!
let x = 68;

// use names that describe the value they contain
let currentTemp = 68;

// it's ok to use short variable names as counters, like i in a for loop
for (let i = 1; i <= 3; i++) {
  console.log(i);
}

```