FUNCTIONAL PROGRAMMING

Technical Communication

FORMAT

- Question prompt
- 5 minutes to research / discuss with partner
- 5 minutes to interview each other
- Share thoughts with the class

Question 1

- Explain currying.
- What are situations where you might use it?

- Currying is a process in functional programming in which we can transform a function with multiple arguments into a sequence of nesting functions that take fewer arguments.
- Currying always returns another function with only 1 argument until all of the arguments have been applied.

```
function multiply(a) {
    return (b) => {
        return (c) => {
            return a * b * c
        }
    }
}
log(multiply(1)(2)(3)) // 6
```

Example use case:

- Redux thunks
- Event handlers

Question 2

- What is closure?
- How do you use it in JavaScript?

Closure

Feature in JavaScript where where an inner function has access to the outer function's variables.

```
1   function outerFunction(a){
2     let sum = a
3     return (num) => sum + num
4   }
5
6   let adder = outerFunction(1)
7
8   adder(3) //=> 4
9
10
```

Why do you use it?

- Commonly used to give objects data privacy.
- A **closure** gives you **access** to an outer function's scope from an inner function.
- Closures are used for partial application and currying

Question 3

- What is statically typed Language?
- What is the benefit of statically typed Languages?
- Given that, why choose JavaScript?

Statically typed languages

- STL do **type** checking (i.e. the process of verifying and enforcing the constraints of types) at compile-time as opposed to run-time.
- For some languages, this means that you as a programmer must **specify** what **type** each variable is (Java, C, C++, TypeScript)

Statically Typed Benefits

- Better code completion
- **Static typing** makes it easier to work with relational DBs
- Main advantage is that all kinds of checking can be done by the compiler, and therefore a lot of trivial bugs are caught at a very early stage

Why choose JS?

- Dynamic types make code flexible
- Variables' types are *dynamic*, meaning after you set a variable to a type, you CAN change it.
- No compilation step, can immediately run/test code


YOU COME TO ME AT RUNTIME
TO TELL ME THE CODE YOU ARE EXECUTING DOES NOT COMPILE
memegenerator.net