

Quiz Practice

Unit 1 Review

Topics

- git Branching, Merging, and Remotes
- Angular Fundamentals (Services, Components)
- Closures, Rest vs. Spread Syntax, Destructuring Assignment
- Dependency Injection
- Metaprogramming with Decorators and @Annotations
- HTTP
- FastAPI Fundamentals from EX03

Dependency Injection

- Describe the general concept of *dependency injection* at a high level. What benefits does it provide to a system? Give an example of how it is used in Angular or FastAPI.

Git Merging, Branching, and Remotes Review

- What is the fundamental difference between a commit and a merge commit?
- Assume diverging histories on these branches, but no conflicts! Draw the simplest diagram you can of what happens when you run the following commands (assume neither results in an error) and that your starting branch is `main`:
 - `git switch foo`
 - `git merge bar`
- What happens when you **add** a remote to a git repository? What happens when you **fetch** from it?

Closures: Diagram the Following Code Listing

```
1  from typing import Callable
2
3
4  def times(n: int) -> Callable[[int], int]:
5      def times_n(x: int) -> int:
6          return x * n
7
8      return times_n
9
10
11 double = times(2)
12 print(double(3))
```

VarArgs/Rest Syntax vs. Spread Syntax

```
1  const f = (y: number, ...xs: number[]) => {  
2  |    console.log(xs.map((x) => x + y));  
3  };
```

Write two function calls to f, the first should **not use spread syntax, the second **must** use spread syntax. You can define additional variables if useful.**

Metaprogramming

@Decorators are used for *metaprogramming* in both Angular and FastAPI. Where have you seen them used in each framework? What do these uses have in common?

FastAPI and HTTP

- What is the difference between path and query parameters?
- What HTTP method(s) do *not* have a request body?