SD: Design Principles: Part VI

Total points 10/10

Email *

emmauyentrinh@gmail.com

✓ The LSP principle directly addresses issues with * 2/2

- ○ abstraction
- ○ encapsulation
- ◉ inheritance ✓
- ○ polymorphism

✓ The LSP principle applies only to compile time checks and not runtime verifications. *2/2

- ○ yes
- ◉ no ✓

✓ When overriding a method, we need to ensure that we preserve the advertised: *2/2

- ○ code quality
- ◉ requirements and promises ✓
- ○ requirements and polymorphism
- ○ promises and polymorphism

✓ If a derived class inherits from a base, then * 2/2

- [x] every instance of derived is also an instance of base ✓
- [x] what is true of base is true of derived ✓
- [ ] what is true of derived is true of base
- [x] anywhere an instance of base is expected, we should be able to pass an instance of derived ✓

✓ Inheritance: * 2/2

- [x] promotes reuse ✓
- [x] increases coupling ✓
- [x] is often misused ✓
- [x] can be hard to implement correctly ✓