LaMarcus Williams

Research

5/1/2023

Prompts:

What are the four pillars of Object-Oriented Programming? Explain each pillar.

1) Encapsulation: This principle is about the hiding of excess internal code and utilizing only what is necessary to make the code look as clean as possible.
2) Abstraction: This principle is about simplifying the method/system into smaller parts. This is necessary to help with usage of the code without changing other parts of the code.
3) Inheritance: This principle is about creating new classes based off of existing ones and "inheriting" the properties from the parent class. This helps make the code easier to reuse and create new class with similar functions.
4) Polymorphism: This principle is about using objects interchangeably while also maintain the same method or properties. Polymorphism is executed by using inheritance and function overloading.

What is the relationship between a Class and an Object?

The relationship between a class and an object is the class is a blueprint for creating an object while the object is instance of the class. An analogy that helps me understand this is when referring to the NFL, the Seattle Seahawks would be the class and the players, coaches, plays on that team would be objects within the class. The class is more of the structure and the functions of the set of objects while an object is specific to the class with its own functions and properties.

What is an exception and what are best practices for handling them?

An exception, is an error that happens when executing line of code. The types of errors that can occur are syntax, runtime and logical errors within the line of code. Some methods in handling them are using try-catch blocks, throw statements and try-catch-finally statements. The most common way being the try-catch.

Urls:

The 4 Pillars of OOP in JavaScript: A Complete Guide - KudingoBhai

The Four Pillars of Object-Oriented Programming (freecodecamp.org)

8.1 Java | Class & Object Relationship - The Revisionist

JavaScript try-catch - javatpoint

Exception Handling in JavaScript - javatpoint