Research Week 5

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

The four main pillars of Object-Oriented Programming are Polymorphism, Inheritance, Encapsulation, and Abstraction. These ideas are an attempt to implement physical or real-world items in programming languages.

Abstraction:

The concept of abstraction is to present to the user only the information they need to know. The user will see only the essential features and functionality of an object, while the complex code that makes the program function is hidden from the user. The user does not need to know how to calculate their taxes, only how much they are required to pay.

Encapsulation:

Encapsulation entails controlling access to the fields and methods of a class or object. The fields, or attributes of the object, are made private so they can only be accessed within the class. The ability to modify or see the fields is provided by methods called setters and getters, which are set to public access. The data of the class or object is considered to be wrapped up into a single package, like a capsule around medicine.

Inheritance:

Inheritance is the ability to have one class or object take (inherit) some or all the properties and characteristics of another class or object. This allows the reusability of code without having to duplicate code for classes with similar functions. The class that lends its properties to another class is called the Super or Parent class. The class or classes that inherit from the Parent class is called the Child class or subclass. If a child class needs to have some different functionality added later, instead of changing an existing child class, a new child class can be created with the new functionality without disrupting the existing code.

Polymorphism:

The basic definition of polymorphism is “many forms.” This concept means that a function can be performed in different ways by different subclasses and the parent class. For example, cows, alligators, and birds are all animals, so they could be subclasses of the animal class, but each type of animal has its own diet, so when the user requests information about their diet, each animal object would use the Diet method to respond, but each response would be different from the others.

References

Four Main Object Oriented Programming Concepts of Java. (2022, September 15). Retrieved December 10, 2022 from https://www.geeksforgeeks.org/four-main-object-oriented-programming-concepts-of-java/

4 Pillars for Object Oriented Programming. (2019, February 02). Retrieved December 10, 2022 from https://www.linkedin.com/pulse/4-pillars-object-oriented-programming-pushkar-kumar/

Research Week 5

6. What is your favorite thing you learned this week?

I enjoyed most this week learning about Interfaces. I have had classes in Java before and have been exposed to the concept of interfaces, but I never really grasped the advantages of using interfaces. Now I have a much better understanding of how they work and where they can be beneficial. As we were told in the video lesson on interfaces, we should program to an interface and try to incorporate all the methods the interface will need. Then we can create the implementing classes based on the interface. This allows the programmer to create a new class implementing the interface if new functionality is required without changing existing code and possibly breaking code downstream, like adding more bricks when your Lego Eiffel Tower needs more support to keep it from collapsing instead of trying to replace the existing bricks.