Intro to Java Week 5 Research Assignment

**Points possible:** 30

|  |  |  |
| --- | --- | --- |
| Category | Criteria | % of Grade |
| Accuracy | Is the information accurate? | 25 |
| Organization | Is the essay clean and organized? Ideas are presented in a logical order. | 25 |
| Citations | Students reference and cite at least 5 sources. | 25 |
| Completeness | All requirements of the assignment are complete. | 25 |

**Instructions:** In however many words necessary, write a thorough essay response to each of the below prompts. Be sure to include at least 5 references for this assignment. Do not copy and paste text from the internet or any other source; use the information you find in your research, summarize, in your own words, the concepts. Plagiarism will result in a zero for the assignment as well as disciplinary actions. Push this document to your GitHub repository for this week. Add the URL for this week’s repository to this document where instructed and submit this document to your instructor when complete.

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

1. Abstraction: it the process in java which is used to hide certain details and only show essential features of the object. essential features we know about. Ex: triangle, table (your videos).
2. Encapsulation: the wrapping up of data under a single unit. It is a protective shield that prevents the data from being accessed by the code outside this shield. Ex: driving a car you can control where it goes and how fast it goes but not how the cylinders fire. ( your helpful video).
3. Inheritance: is a mechanism in which one object acquires all the properties and behavior of a parent object. This is so you can create new classes that are built upon existing classes. (your videos).
4. Polymorphism: is a concept by which we can perform a single action in different ways. So polymorphism means many forms. Ex: communication between animals. (you video).

**What is the relationship between a Class and an Object?**

Class is how we structure out code in OOP, class is like a blueprint. Object is the actual project developed from the blueprint, an object is an instance of a class. (your videos).

**What are the differences between checked and unchecked exceptions?**

Checked exception: Are the exception checked at compile time, if some code within a method throws a checked exception then the method must either handle the exception or it must specify the exception using the throws key. (your videos).

Unchecked exception: are the exceptions not checked at compile time. this one is in out control and only happens when we make a mistake.

**What are the differences between abstract classes and interfaces? When should you use one over the other?**

An interface is a contract so one person writes the interface and says it will look this way and another person who uses the interface says yeah this is fine if it looks that way. An interface is an empty shell it can’t do anything it’s just a pattern (1.). Abstract classes are classes not like interfaces you can define a behavior from a abstract class, “these classes should look like that, and they have that in common, so fill in the blanks.” (1). You would use and abstract calls when you want to inherit something from another code.

**What is unit testing and why is it important?**

When you unit test you break your program into pieces and subject each small piece in to a series of tests. You do this so you can catch any problems before they get to out of hand.

**What is your favorite thing you learned this week?**

I love how we can run code and actually add names or information. It feels like we are becoming better at this.

**References:**

1. [**https://stackoverflow.com/questions/1913098/what-is-the-difference-between-an-interface-and-abstract-class**](https://stackoverflow.com/questions/1913098/what-is-the-difference-between-an-interface-and-abstract-class)
2. [**https://stackoverflow.com/questions/652292/what-is-unit-testing-and-how-do-you-do-it**](https://stackoverflow.com/questions/652292/what-is-unit-testing-and-how-do-you-do-it)

**URL to GitHub Repository:**