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

   The Four Pillars of object-oriented programing are polymorphism, abstraction, encapsulation & inheritance.

   Polymorphism –allows you do refer to different types of objects at different points of time. Think of method overloading

   Abstraction – hides the complexity of how data is calculated from the user and only shows what they need.

   Encapsulation – only gives access to the variables the user needs to change. The rest are calculated within methods or at declaration

   Inheritance – is to get or inheritate existing attributes from the parent class

2. What is the relationship between a Class and an Object?

   Class is the blueprint of the object. Object is an instance of the class.

3. What are the differences between checked and unchecked exceptions?

   Checked exceptions are found when we compile the program. Unchecked exceptions are found when the program runs.

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

   Abstract classes can have abstract methods and non-abstract. However, interfaces can only have abstract methods. When using either, abstract methods need to be created in the class you are making. Which one depends on if you will have common code. WE should also write our code for reusability. So if we can reuse things use abstract class but if there is no common code then interface

https://blog.knoldus.com/understanding-the-concept-of-oop-its-four-pillars/

https://www.geeksforgeeks.org/checked-vs-unchecked-exceptions-in-java/

https://www.geeksforgeeks.org/difference-between-abstract-class-and-interface-in-java/