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

The four pillars are Abstraction, Encapsulation, Inheritance, Polymorphism. Abstraction is a way to hide the details to avoid duplication of code but it can also increase its usability by making it simpler and easier to reuse. Encapsulation is when longer bits of code are broken down into smaller functions where the data can be stored and accessed as needed without change or damage, if a process isn't needed often then it can be encapsulated in a way where it isn't accidently brought up while a more needed process runs and vice versa. Inheritance is when one object, class, function etc. acquires properties from another higher up object. It makes it so programmers can create a parent which future objects can pull data from as needed so they do not need to rewrite the same attributes for multiple objects. Polymorphism is related to inheritance in how it uses previous code to perform other like tasks. The data from the parent is used for the base but then expanded to include more "sub-class" information.

Resources:

https://www.freecodecamp.org/news/four-pillars-of-object-oriented-programming/

https://stackoverflow.com/questions/27642239/what-is-polymorphism-in-javascript

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

Exceptions are errors or abnormal conditions in the code which will need special techniques. It is a bug or error which causes the process to break and the code to fail. JavaScript has keywords used in this process- throw, try, catch, finally are used to 'troubleshoot' possible errors. Try allows you to try a segment of code for test for errors and catch allows you to define what happened is the try finds an error. Throw is what normally happens when an error in JavaScript occurs but can also be forced by the user and set to throw an exception. Throw can also be used with the try and catch to force an exception. The finally statement is used after the try and catch keys to force execute the code even if there is an exception or error.

Resources:

https://www.w3schools.com/js/js_errors.asp