Design patterns in JS

http://www.dofactory.com/javascript/singleton-design-pattern

what is design pattern??

Every developer strives to write maintainable, readable, and reusable code.

Code structuring becomes more important as applications become larger.

Design patterns prove crucial to solving this challenge – providing an organization structure for common issues in a particular circumstance.

Types :

Creational - mainly abt creation of objects

CONSTRUCTOR, FACTORY, Abstract, PROTOTYPE, SINGLETON and Builder.

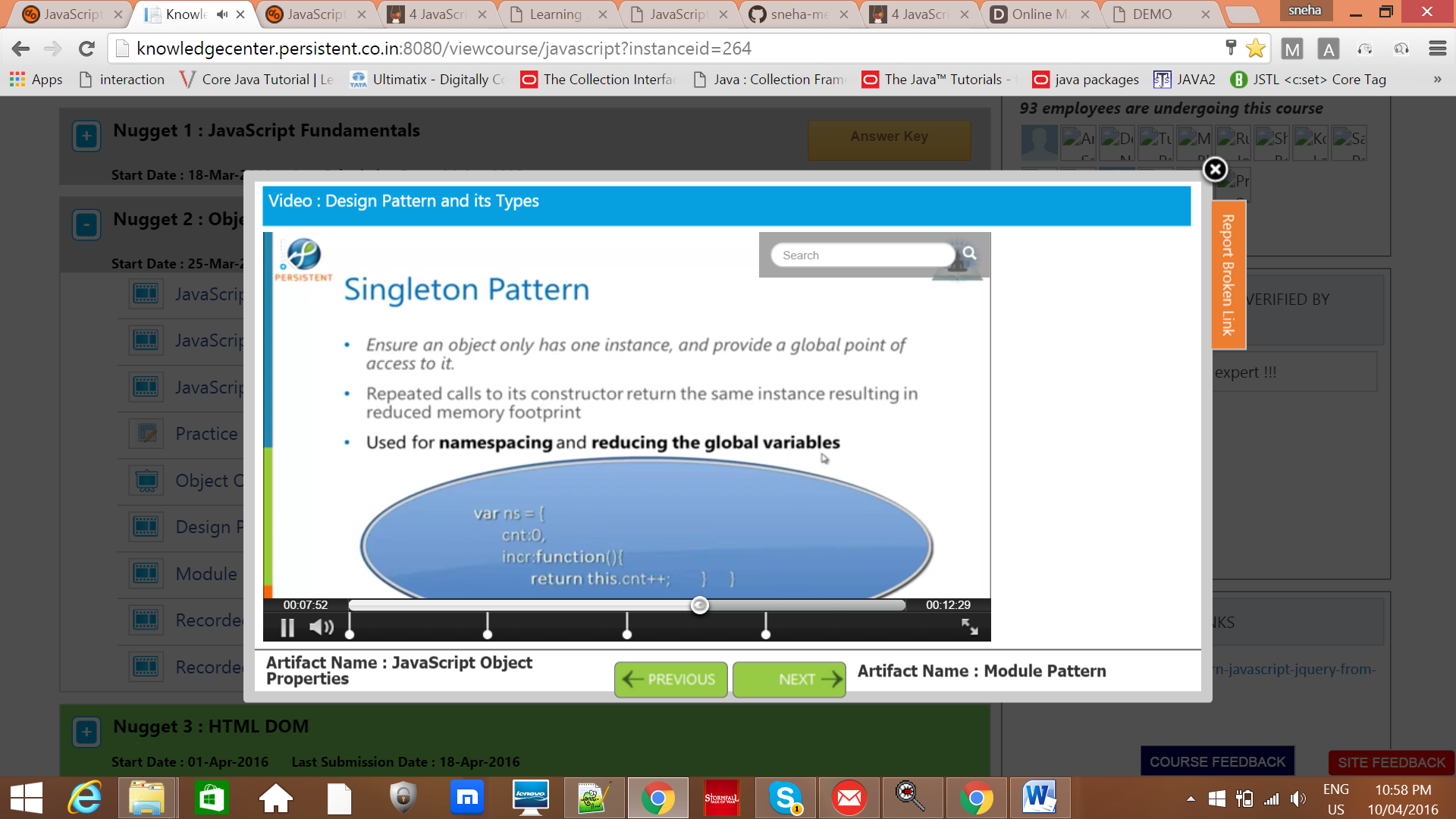
Structural - Structural patterns are concerned with object composition and typically identify simple ways to realize relationships between different objects. They help ensure that when one part of a system changes,the entire structure of the system doesn't need to do the same.

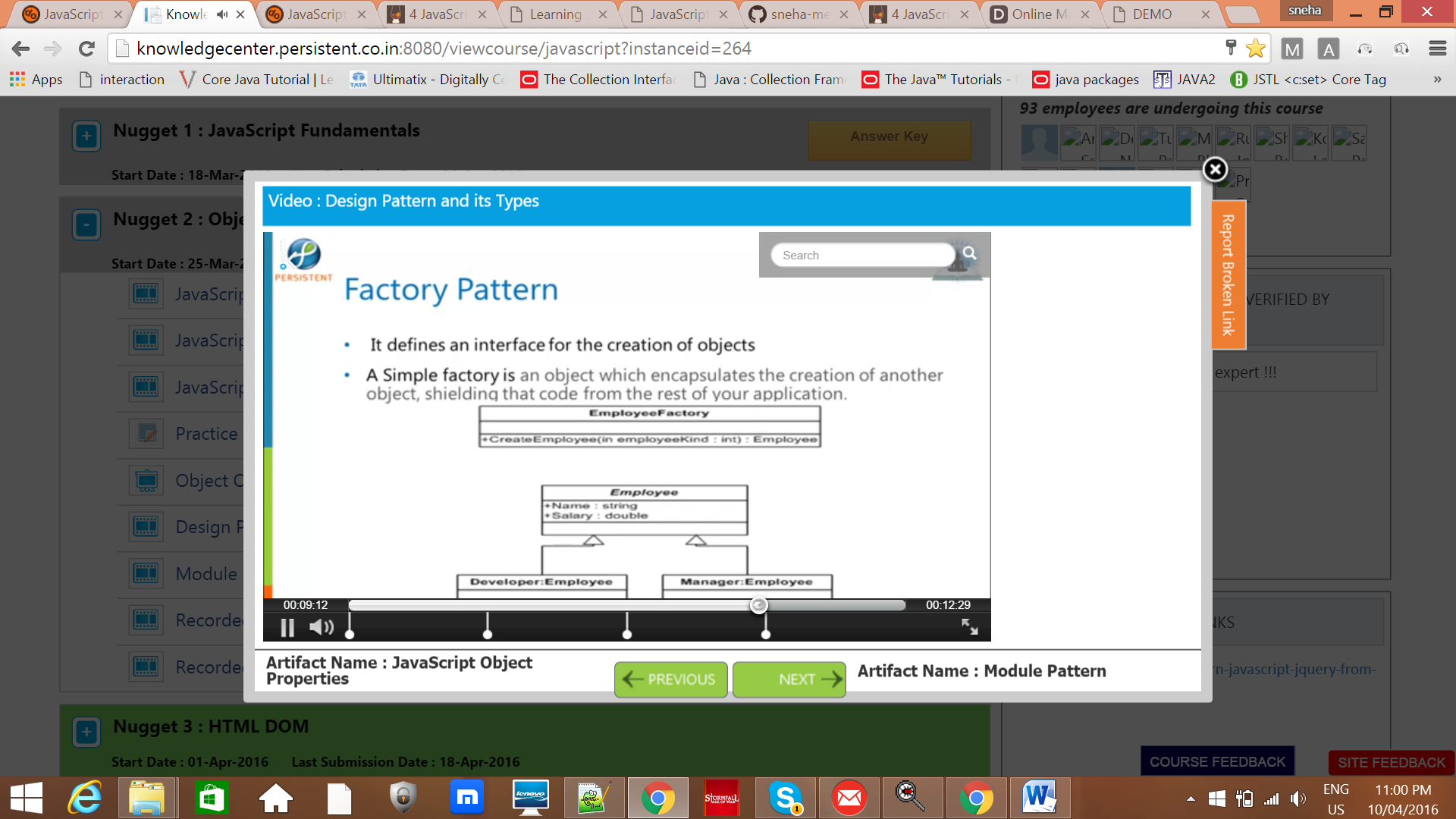
eg: Decorator, FACADE, Flyweight, Adapter and Proxy,command

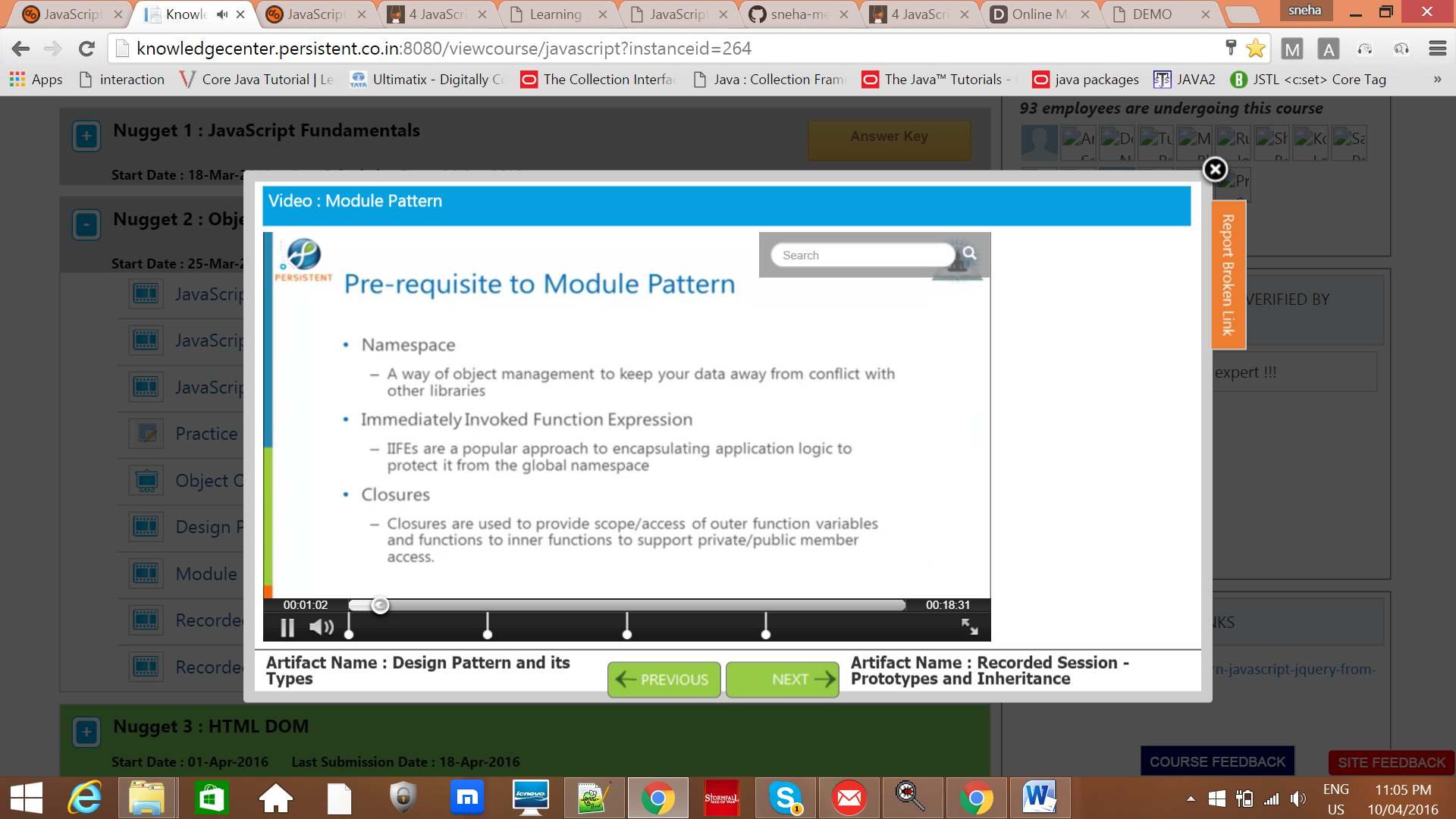
Behavioral - Behavioral patterns focus on improving or streamlining the communication between disparate objects in a system.

eg: Iterator, Mediator, OBSERVER and Visitor.

<http://knowledgecenter.persistent.co.in:8080/viewcourse/javascript?instanceid=264>







Factory pattern :

The factory gets its name from its intended purpose, simplifying the creation of objects. Simple factories abstract out all of those uses of the new keyword so that if a class name changes or is replaced by a different one, you only need to change it in one place. Plus it sets up a one-stop-shop for creating many different types of objects or a single type of object with different options. The standard factory is a little tougher to explain in so few words, so I bring that up later.