**W04 Forms, Object Oriented Programming, Modular JavaScript**

**Chapter 8, Forms**

[**https://www.sitepoint.com/premium/books/javascript-novice-to-ninja-2nd-edition/read/8**](https://www.sitepoint.com/premium/books/javascript-novice-to-ninja-2nd-edition/read/8)

*Form Syntax*

Graphical user interface, text

Description automatically generated with medium confidence

Reset buttons are considered poor for usability.

**Chapter 12, Object-Oriented Programming**

<https://www.sitepoint.com/premium/books/javascript-novice-to-ninja-2nd-edition/read/12>

*Encapsulation –* Keep all programming logic inside an object and make methods available to implement the functionality, without the outside world needing to know *how* it’s done.

*Polymorphism –* Various objects can share the same method, but also have the ability to override shared methods with a more specific implementation.

*Inheritance –* Take an object that already exists and inherit all its properties and methods. We can then improve on its functionality by adding new properties and methods.

JavaScript is prototype-based, which means its classes are actual objects.

*Constructor functions*

Graphical user interface

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

Graphical user interface

Description automatically generated

You can use Prototype inheritance to change a class of an object in the middle of a script!

*Public and private methods*

Public means they can be changed, private means they can’t.

Graphical user interface, application, Teams

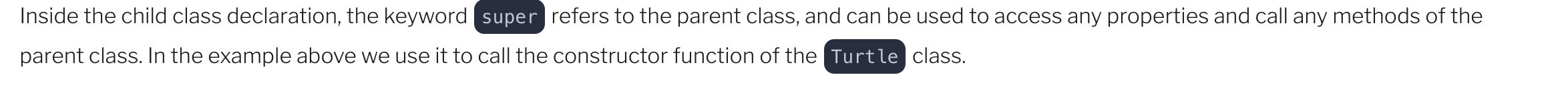
Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, website

Description automatically generated



*Getters and Setters*

Text

Description automatically generated

*Creating objects from objects*

Graphical user interface, website

Description automatically generated

You can add properties and methods to objects using the mixin() function.

**Chapter 15: Modern JavaScript Development**

*Modular JavaScript*

* Keeping code organized in separate, reusable files.
* Code in a module should have a single purpose.
* Allows a public API to be exposed while keeping the implementationhidden inside the module.

Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, text

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

*Graphical user interface, application, Teams

Description automatically generated*

*Graphical user interface, text

Description automatically generated*

*Graphical user interface, application

Description automatically generated*

Note that modules only work via HTTP. Use a server to work with them.

*Applications for work*

* Use .focus() to focus a field in a form. I can make the first field in the form automatically focused.
* I can also use the placeholder attribute in html to give fields placeholder text. This way I can give hints on how to fill out a form.
* Text, application

  Description automatically generated

Text

Description automatically generated

* For dropdown menus, I can have a default value with no value for the help text.

Text

Description automatically generated

* I can write a script to validate fields. For example, it can validate telephone numbers, email addresses, etc.
  + I can validate fields by inserting another label below the other labels

Text

Description automatically generated\

* If there is an error on the form, I can disable the next step button. Just add the disabled attribute to the button.

*Questions*

* Is PHP still widely used? Are there better server-side alternatives?