Iterables and Iterator

1. **Iterables** protocol allows an object to define or customize their iteration behavior.

2. To be an Iterable, an object **must implement** the @@iterator method. Meaning that the object must have a property with @@iterator key which is available via constant **Symbol.iterator**

3. [Symbol.iterator] is a zero arguments function that **returns** an object , conforming to **iterator** protocol.

4. Whenever an object needs to be iterated (such as for…of loop), its @@iterator method is called with no arguments, and the returned iterator is used to obtain the value to be iterated.

5. Iterator protocol defines a standard way to produce a sequence of values and potentially a return value when all values have been generated.

6. An object is an **Iterator** when it **implements** the **next**() function.

7. Next() function is a zero arguments one that **returns** an **object** with 2 keys (**done**: true/false and **value**: returned value)

To learn more and get OneNote, visit
https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Iteration_protocols