Name:____________Qixiang Liu_____________________

Eloquent JavaScript

EECS 368 sp'18 Chapters 8 & 10 (Bugs & Errors + Modules)

Due: Friday 2nd March, 10am (start of class)

Instructions: Read Chapter 8, and Chapter 10 of Eloquent JavaScript THIRD EDITION (**http://eloquentjavascript.net/3rd_edition/**) and complete the exercises listed below and submit your answers on **Blackboard.**
Just a reminder to take the time to understand **all** the code throughout the readings or things in class probably won't make sense and you'll fall behind. You're in charge of your own education.

1. By signing (or e-signing) below, you attest that you have read all sections of this reading assignment.
   Qixiang Liu
2. List *at least* **3** quotes from the reading that you thought were particularly meaningful. For example, one of mine is: "There are those who will say *terrible* things about the JavaScript language. Many of these things are true." --(Introduction, Eloquent JS)
   a. It is better to report to the user what went wrong and then give up. –debugger
   b. An important part of programming is finding, diagnosing, and fixing bugs.
   c. Write code that is easy to delete, not easy to extend.

3.
   a. At the end of Chapter 8:
      i. Complete at least one of the exercises.
   b. At the end of Chapter 10:
      i. Complete at least one of the exercises.

Note that any time an exercise lists something as extra(or similarly worded), you must complete as part of the exercise.
There is a sandbox to run javascript code directly on eloquentjavascript.net (http://eloquentjavascript.net/code/). Submit which problems you do and *your* code for them.

1.Retry:

```
class MultiplicatorUnitFailure extends Error {}

function primitiveMultiply(a, b) {
 if (Math.random() < 0.2) {
   return a * b;
 } else {
   throw new MultiplicatorUnitFailure("Klunk");
 }
}
function reliableMultiply(a, b) {
 try{
	return primitiveMultiply(a,b);
 }catch(err){
	if(err instanceof MultiplicatorUnitFailure){
	  return primitiveMultiply(a,b);
  }else{
          return err;
  }
 }
}
console.log(reliableMultiply(8, 8));
// → 64
```

2.Chapter 10: cyclic dependencies:

```
//a.js
var ClassB = require(“./b”);
console.log(‘class b: ’,ClassB);
var ClassA = function(){
        this.thing = ClassB();
        this.property = 5;
}
var a = new ClassA();
module.exports = a;
//b.js
var ClassB = function(){}
classB.propotype.doSomthingLater(){
        console.log(a.property);
}
module.exports = ClassB
var a = require(“./a”);
```

4. Since this is a draft edition, try to catch any errors you find and report them (if not already reported). This book is made available for free; the least we could do is help make it better! Summarize any errors you find (it’s okay if you don’t find any, but indicate so)