1. Numbers: NEVER USE QUOTES for a number if you want to do math on it!

```
var money = 100;
var tempInAlaska = -32.4;
var todaysMonth = 2;
```

2. Strings: use quotes on anything that you want it to be considered as string, especially anything contains not just numbers.

```
var moneyString = "$100.00";
var temp = "32.4 degrees";
var todaysMonthString = "Feb.";
```

3. Conditionals:

less than --- <   greater than --- >   less than or equal --- <=   grater than or equal --- >=

equal --- ==   not equal --- !=

var a=100; // A single equal is an assignment, which means set the value of a to 100.

If a == 100   // A double equal is a comparison, which means the condition is a equals 100.

```
1   var a = 20;
2   a = a + 10;
3   var b = 10;
4   b = a + b;
5
6   if (a > b) { console.log("A"); }
7   else if (b > a) { console.log("B"); }
8   else { console.log("AB"); }
9
```

```
Native Chrome JavaScript.
Copyright (c) 2013 Google Inc
>
B
>
```

4. Fountions: You can define function by creating variables and assigning them the value of a function.

1) We defined a function named printHello, and the usage of the function is to print "Hello" "CS" "105"on the screen by using consle.log. When we call the function for four times, it will execute what we just set for four times.

```
1    var printHello = function()
2 ▾  {
3       console.log("Hello");
4       console.log("CS");
5       console.log("105");
6    }
7    printHello();
8    printHello();
9    printHello();
10   printHello();
11
```

```
Native Chrome JavaScript.
Copyright (c) 2013 Google Inc
>
Hello
CS
105
Hello
CS
105
Hello
CS
105
Hello
CS
105
>
```

2) Variable a and b inside the parenthesis are called functions parameters. As they are naturally considered as variables, we do not have to define them by writing var a and var b again. In the 6th line, we set a=10 and b=5, so c=15; in the 7th line, we set a=c and b=10, so d=15+10=25.

```
1   var sum = function(a, b)
2 ▾ {
3     var c = a + b;
4     return c;
5   }
6   var c = sum(10, 5);
7   var d = sum(c, 10);
8   console.log(d);
```
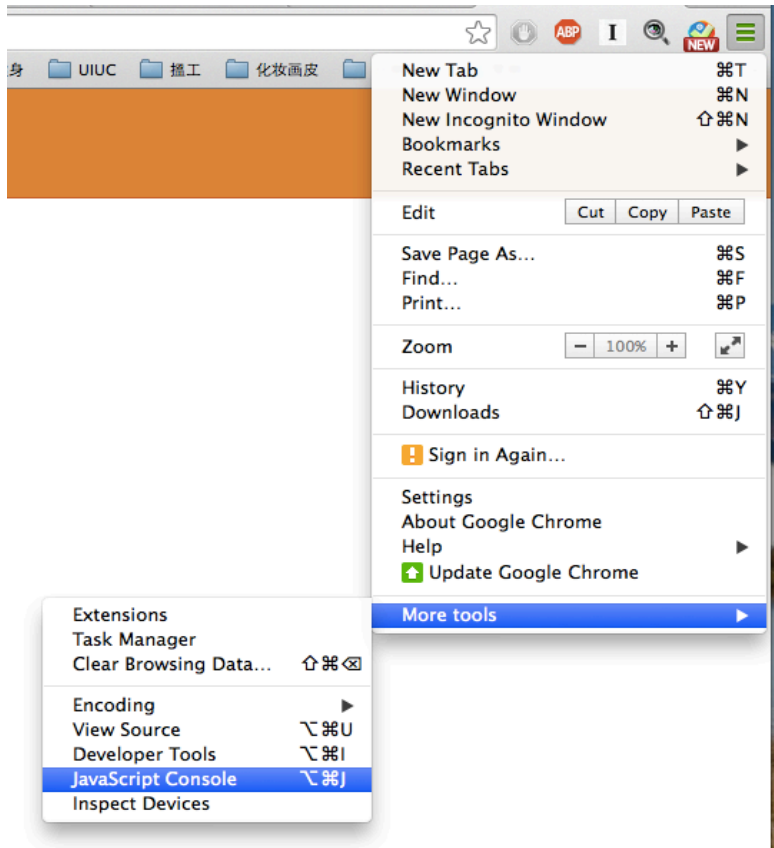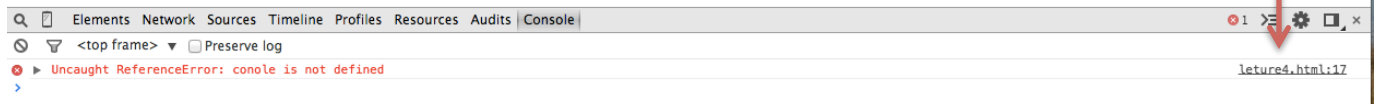
```
Native Chrome JavaScript.
Copyright (c) 2013 Google Inc
:>
25
:>
```

5. How to use the web browser console

For Firefox: Double click to open lecture4.html, Developer --- Browser Console

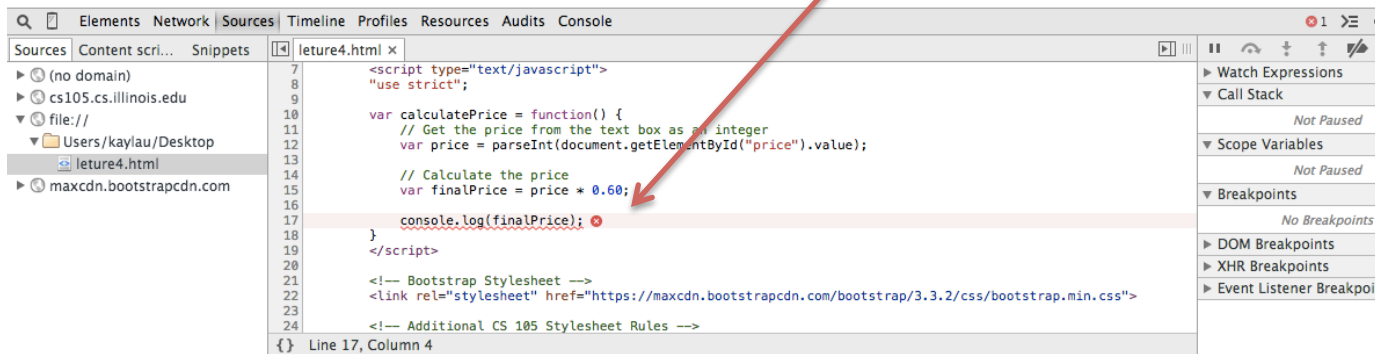For Google Chrome: Double click to open lecture4.html, more tools --- JavaScript Console

If there is something wrong with your code, you will find the error. Click the location of the error, you are able to go to the exact location of the error in sources:

Lecture 4: Functions

Discount Calculator

Original Price: 20   Calculate!