

# CS 105

**Week 3**

MP1

Due **tomorrow** by 6:00pm

**MP: Explained Session**  
**MPx Session for MP1**

Tomorrow, Tuesday, Sept. 16<sup>th</sup>  
7:00pm, 0216 Siebel Center

# Weekly Activity 4

Out: Today

Due: Monday, Sept. 22<sup>nd</sup> by 9:00am

i>Clickers

```
print("Enter the prices of  
      your items:");  
var total = 0;  
  
var price = readline();  
total = total + (price * 1);  
  
while (x != 0)  
{  
    price = readline();  
    total = total + (price * 1);  
}
```

# Functions

```
function readnumber()  
{  
    var s = readline();  
    return (s * 1);  
}
```

```
function readnumber()  
{  
    var s = readline();  
    return (s * 1);  
}
```

Numbers

# Numbers

Integers (“ints”)

Floating Point Numbers (“floats”)

# Integers

Any whole number, either positive, negative, or zero.

10

223

1

54248

11886

-3

-4921

-21

0

# Floats

Any non-integer, real number.  
(As best as the computer can do.)

0.125

-48.5

0.5

-45.15412

0.1111111

1.85

3.3333

9.12345

-800.001

`parseFloat()`

```
function readnumber()  
{  
    var s = readline();  
    return (s * 1);  
}
```

```
function readnumber()  
{  
    var s = readline();  
    return parseFloat(s);  
}
```

From `parseFloat()` documentation:

If the first character cannot be converted to a number, `parseFloat` returns **NaN**.

For arithmetic purposes, the **NaN** value is not a number in any radix. You can call the `isNaN` function to determine if the result of `parseFloat` is **NaN**. If **NaN** is passed on to arithmetic operations, the operation results will also be **NaN**.

```
function readnumber()  
{  
    var s = readline();  
    var f = parseFloat(s);  
    if (isNaN(f) == true)  
    {  
        print("Try again:");  
        s = readline();  
        f = parseFloat(s);  
    }  
    return f;  
}
```

```
function readnumber()  
{  
    var s = readline();  
    var f = parseFloat(s);  
    while (isNaN(f) == true)  
    {  
        print("Try again:");  
        s = readline();  
        f = parseFloat(s);  
    }  
    return f;  
}
```

Making Change

**Total : \$4.33**

**Total : \$4.33**

**Cash : \$5.00**

Total : \$4.33

Cash : \$5.00

Change : \$0.67