iClicker Experiment
iClicker Experiment

Throughout the lecture, update me on how you’re doing
iClicker Experiment

Throughout the lecture, update me on how you’re doing

A "I’m really happy right now. This is easy"
B "That totally makes sense when I think about it"
C "This is challenging, but I can do it"
D "I am pretty confused"
E "I really wish I weren’t here right now"
A quick Review of Data Types
A quick Review of Data Types

Variables hold data

Numbers for example

Or Strings
A quick Review of Data Types

There are many forms of data
There are many forms of data...

Images

Sound Files

Videos

etc.
How are those stored?
How are those stored?

Numbers
   Not terribly hard

Everything Else
   A bit more complex
How are those stored?

Numbers
Not terribly hard

Everything Else
A bit more complex

We know how that works!
How are those stored?

Numbers
Not terribly hard

Everything Else
A bit more complex

Now let's do this!
Storing Strings
Storing Strings

Numbers are easy to store
Storing Strings

Numbers are easy to store

So let's just translate a string into a number and store that
Storing Strings

For that we use a table.
Storing Strings

For that we use a table.

Like ASCII for example.
<table>
<thead>
<tr>
<th>Decimal</th>
<th>Hex</th>
<th>Char</th>
<th>Decimal</th>
<th>Hex</th>
<th>Char</th>
<th>Decimal</th>
<th>Hex</th>
<th>Char</th>
<th>Decimal</th>
<th>Hex</th>
<th>Char</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>[NULL]</td>
<td>32</td>
<td>20</td>
<td>[SPACE]</td>
<td>64</td>
<td>40</td>
<td>@</td>
<td>96</td>
<td>60</td>
<td>`</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>[START OF HEADING]</td>
<td>33</td>
<td>21</td>
<td>!</td>
<td>65</td>
<td>41</td>
<td>A</td>
<td>97</td>
<td>61</td>
<td>a</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>[START OF TEXT]</td>
<td>34</td>
<td>22</td>
<td>&quot;</td>
<td>66</td>
<td>42</td>
<td>B</td>
<td>98</td>
<td>62</td>
<td>b</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>[END OF TEXT]</td>
<td>35</td>
<td>23</td>
<td>#</td>
<td>67</td>
<td>43</td>
<td>C</td>
<td>99</td>
<td>63</td>
<td>c</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>[END OF TRANSMISSION]</td>
<td>36</td>
<td>24</td>
<td>$</td>
<td>68</td>
<td>44</td>
<td>D</td>
<td>100</td>
<td>64</td>
<td>d</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>[ENQUIRY]</td>
<td>37</td>
<td>25</td>
<td>%</td>
<td>69</td>
<td>45</td>
<td>E</td>
<td>101</td>
<td>65</td>
<td>e</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>[ACKNOWLEDGE]</td>
<td>38</td>
<td>26</td>
<td>&amp;</td>
<td>70</td>
<td>46</td>
<td>F</td>
<td>102</td>
<td>66</td>
<td>f</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>[BEL]</td>
<td>39</td>
<td>27</td>
<td>'</td>
<td>71</td>
<td>47</td>
<td>G</td>
<td>103</td>
<td>67</td>
<td>g</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>[BACKSPACE]</td>
<td>40</td>
<td>28</td>
<td>(</td>
<td>72</td>
<td>48</td>
<td>H</td>
<td>104</td>
<td>68</td>
<td>h</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>[HORIZONTAL TAB]</td>
<td>41</td>
<td>29</td>
<td>)</td>
<td>73</td>
<td>49</td>
<td>I</td>
<td>105</td>
<td>69</td>
<td>i</td>
</tr>
<tr>
<td>11</td>
<td>B</td>
<td>[VERTICAL TAB]</td>
<td>43</td>
<td>2B</td>
<td>+</td>
<td>75</td>
<td>4B</td>
<td>K</td>
<td>107</td>
<td>6B</td>
<td>k</td>
</tr>
<tr>
<td>12</td>
<td>C</td>
<td>[FORM FEED]</td>
<td>44</td>
<td>2C</td>
<td>,</td>
<td>76</td>
<td>4C</td>
<td>L</td>
<td>108</td>
<td>6C</td>
<td>l</td>
</tr>
<tr>
<td>13</td>
<td>D</td>
<td>[CARRIAGE RETURN]</td>
<td>45</td>
<td>2D</td>
<td>`-</td>
<td>77</td>
<td>4D</td>
<td>M</td>
<td>109</td>
<td>6D</td>
<td>m</td>
</tr>
<tr>
<td>14</td>
<td>E</td>
<td>[SHIFT OUT]</td>
<td>46</td>
<td>2E</td>
<td>,</td>
<td>78</td>
<td>4E</td>
<td>N</td>
<td>110</td>
<td>6E</td>
<td>n</td>
</tr>
<tr>
<td>15</td>
<td>F</td>
<td>[SHIFT IN]</td>
<td>47</td>
<td>2F</td>
<td>/</td>
<td>79</td>
<td>4F</td>
<td>O</td>
<td>111</td>
<td>6F</td>
<td>o</td>
</tr>
<tr>
<td>16</td>
<td>10</td>
<td>[DATA LINK ESCAPE]</td>
<td>48</td>
<td>30</td>
<td>0</td>
<td>80</td>
<td>50</td>
<td>P</td>
<td>112</td>
<td>70</td>
<td>p</td>
</tr>
<tr>
<td>17</td>
<td>11</td>
<td>[DEVICE CONTROL 1]</td>
<td>49</td>
<td>31</td>
<td>1</td>
<td>81</td>
<td>51</td>
<td>Q</td>
<td>113</td>
<td>71</td>
<td>q</td>
</tr>
<tr>
<td>18</td>
<td>12</td>
<td>[DEVICE CONTROL 2]</td>
<td>50</td>
<td>32</td>
<td>2</td>
<td>82</td>
<td>52</td>
<td>R</td>
<td>114</td>
<td>72</td>
<td>r</td>
</tr>
<tr>
<td>19</td>
<td>13</td>
<td>[DEVICE CONTROL 3]</td>
<td>51</td>
<td>33</td>
<td>3</td>
<td>83</td>
<td>53</td>
<td>S</td>
<td>115</td>
<td>73</td>
<td>s</td>
</tr>
<tr>
<td>20</td>
<td>14</td>
<td>[DEVICE CONTROL 4]</td>
<td>52</td>
<td>34</td>
<td>4</td>
<td>84</td>
<td>54</td>
<td>T</td>
<td>116</td>
<td>74</td>
<td>t</td>
</tr>
<tr>
<td>21</td>
<td>15</td>
<td>[NEGATIVEacknowledge]</td>
<td>53</td>
<td>35</td>
<td>5</td>
<td>85</td>
<td>55</td>
<td>U</td>
<td>117</td>
<td>75</td>
<td>u</td>
</tr>
<tr>
<td>22</td>
<td>16</td>
<td>[SYNCHRONOUS IDLE]</td>
<td>54</td>
<td>36</td>
<td>6</td>
<td>86</td>
<td>56</td>
<td>V</td>
<td>118</td>
<td>76</td>
<td>v</td>
</tr>
<tr>
<td>23</td>
<td>17</td>
<td>[ENG OF TRANSM. BLOCK]</td>
<td>55</td>
<td>37</td>
<td>7</td>
<td>87</td>
<td>57</td>
<td>W</td>
<td>119</td>
<td>77</td>
<td>w</td>
</tr>
<tr>
<td>24</td>
<td>18</td>
<td>[CANCEL]</td>
<td>56</td>
<td>38</td>
<td>8</td>
<td>88</td>
<td>58</td>
<td>X</td>
<td>120</td>
<td>78</td>
<td>x</td>
</tr>
<tr>
<td>25</td>
<td>19</td>
<td>[END OF MEDIUM]</td>
<td>57</td>
<td>39</td>
<td>9</td>
<td>89</td>
<td>59</td>
<td>Y</td>
<td>121</td>
<td>79</td>
<td>y</td>
</tr>
<tr>
<td>26</td>
<td>1A</td>
<td>[SUBSTITUTE]</td>
<td>58</td>
<td>3A</td>
<td>:</td>
<td>90</td>
<td>5A</td>
<td>Z</td>
<td>122</td>
<td>7A</td>
<td>z</td>
</tr>
<tr>
<td>27</td>
<td>1B</td>
<td>[ESCAPE]</td>
<td>59</td>
<td>3B</td>
<td>;</td>
<td>91</td>
<td>5B</td>
<td>[</td>
<td>123</td>
<td>7B</td>
<td>{</td>
</tr>
<tr>
<td>28</td>
<td>1C</td>
<td>[FILE SEPARATOR]</td>
<td>60</td>
<td>3C</td>
<td>&lt;</td>
<td>92</td>
<td>5C</td>
<td>\</td>
<td>124</td>
<td>7C</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>1D</td>
<td>[GROUP SEPARATOR]</td>
<td>61</td>
<td>3D</td>
<td>=</td>
<td>93</td>
<td>5D</td>
<td>]</td>
<td>125</td>
<td>7D</td>
<td>}</td>
</tr>
<tr>
<td>30</td>
<td>1E</td>
<td>[RECORD SEPARATOR]</td>
<td>62</td>
<td>3E</td>
<td>&gt;</td>
<td>94</td>
<td>5E</td>
<td>^</td>
<td>126</td>
<td>7E</td>
<td>~</td>
</tr>
<tr>
<td>31</td>
<td>1F</td>
<td>[UNIT SEPARATOR]</td>
<td>63</td>
<td>3F</td>
<td>?</td>
<td>95</td>
<td>5F</td>
<td>_</td>
<td>127</td>
<td>7F</td>
<td>[DEL]</td>
</tr>
</tbody>
</table>
Let's try that!

How do we store "CS105"?
Let's try that!

How do we store
"CS105" ?
Let's try that!

How do we store "CS105"?
Let's try that!

How do we store "CS105"?

<table>
<thead>
<tr>
<th>C</th>
<th>S</th>
<th>1</th>
<th>0</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>67</td>
<td>83</td>
<td>49</td>
<td>48</td>
<td>53</td>
</tr>
</tbody>
</table>
Let's try that!

How do we store "CS105"?

<table>
<thead>
<tr>
<th>C</th>
<th>S</th>
<th>1</th>
<th>0</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>67</td>
<td>83</td>
<td>49</td>
<td>48</td>
<td>53</td>
</tr>
</tbody>
</table>

| 01000011 | 01010011 | 00110001 | 00110000 | 00110101 |
But what about images?
But what about images?
What a nice blue pixel!
What a nice blueish pixel!
What a nice grey pixel!
What a nice ______ pixel !
What a nice _____ pixel!

<table>
<thead>
<tr>
<th>7050W Blue Chill</th>
<th>7060W Cool Vista</th>
<th>7070W Bridgewater</th>
<th>7080W Sistine Blue</th>
<th>7090W Blue Cool</th>
</tr>
</thead>
<tbody>
<tr>
<td>7051W Blue Bouquet</td>
<td>7061W Soft Sky</td>
<td>7071W Meltwater</td>
<td>7081W Clear Day</td>
<td>7091W Ondine Blue</td>
</tr>
<tr>
<td>7052W Brisk Blue</td>
<td>7062W Carolina Blue</td>
<td>7072W Heron Blue</td>
<td>7082W Sky Delight</td>
<td>7092W Inspiration Blue</td>
</tr>
<tr>
<td>7053M Jonathan</td>
<td>7063M Blue Palace</td>
<td>7073M Bluejay</td>
<td>7083M Blue Yonder</td>
<td>7093M Baby Blue Eyes</td>
</tr>
<tr>
<td>7054M Venetian Blue</td>
<td>7064M Water Jet</td>
<td>7074M Brandon’s Blue</td>
<td>7084M Parade Blue</td>
<td>7094M Blue Stencil</td>
</tr>
<tr>
<td>7055D Pompeii</td>
<td>7065D Triumph Blue</td>
<td>7075D Electron Blue</td>
<td>7085D Atlas Blue</td>
<td>7095D Grandma Blue</td>
</tr>
<tr>
<td>7056N Endless Blue</td>
<td>7066N Stars Forever</td>
<td>7076A Neon Blue</td>
<td>7086A Gulf Coast</td>
<td>7096A Deep Marine</td>
</tr>
</tbody>
</table>
What a nice *Brisk Blue* pixel?
Pretty Colors!
Pretty Colors!

255
0
0
0
Pretty Colors!

255
122
0
Pretty Colors!

255
122
178
Pretty Colors!

0

122

178
Pretty Colors!
Pretty Colors!
What a nice pixel!
What a nice pixel!
What a nice pixel!

168  
192  
215  
10101000  
11000000  
11010111  
168  
215  
192  
11000000  
11010111  
10101000
What a nice pixel!
Iterating over an image

for(x=0;x<theUnion.width;x++){

}
Iterating over an image

```java
for(x=0;x<theUnion.width;x++){
    // Something to do to each pixel
}
```
for(y=0;y<theUnion.height;y++){
    for(x=0;x<theUnion.width;x++){
        // Something to do to each pixel
    }
}

Iterating over an image
Iterating over an image

```java
for(y=0;y<theUnion.height;y++){
    for(x=0;x<theUnion.width;x++){
        thisPixel = theUnion.getRGB(x,y);
    }
}
```
What data type is $thisPixel$?
What data type is `thisPixel`?
It's an "Object".

```javascript
thisPixel = theUnion.getRGB(x,y);
```
What data type is thisPixel?

It's an "Object".

thisPixel
Objects have "Properties"

thisPixel
Objects have "Properties"
Properties are like variables

thisPixel
Objects have "Properties"
Properties are like variables

colorPixel.r
Objects have "Properties"
Properties are like variables

```
thisPixel.r
```

That is a "dot operator"
Objects have "Properties"
Properties are like variables

thisPixel.r
thisPixel.g
Objects have "Properties"
Properties are like variables

thisPixel.r
thisPixel.g
thisPixel.b
Objects have "Properties"
Properties are like variables

alert(thisPixel.r);
thisPixel.g
thisPixel.b
Objects have "Properties"
Properties are like variables

```javascript
alert(thisPixel.r);
thisPixel.g=255;
thisPixel.b=0;
```
Objects have "Properties"
Properties are like variables

Only the object knows its properties. You must use a dot operator to access them!
True of False?
True of False?

You have not used objects or properties in this class before.
True of False?

You have not used objects or properties in this class before.

document   innerHTML
style
Why are Objects special?
Why are Objects special?

for\(y=0; y<\text{theUnion.height}; y++\)\
for\(x=0; x<\text{theUnion.width}; x++\)\
\thisPixel = \text{theUnion.getRGB}(x, y);
Why are Objects special? They also have "methods"!

for(y=0;y<theUnion.height;y++){ for(x=0;x<theUnion.width;x++){ thisPixel = theUnion.getRGB(x,y); } }
Why are Objects special?
They also have "methods"!

for(y=0;y<theUnion.height;y++){
  for(x=0;x<theUnion.width;x++){
    thisPixel = theUnion.getRGB(x,y);
  }
}
Puzzle Time!
Puzzle Time!
If you take this image
Puzzle Time!

If you take this image and apply this code

```java
for(y=0;y<theUnion.height;y++){
    for(x=0;x<theUnion.width;x++){
        thisPixel = theUnion.getRGB(x,y);
        thisPixel.r=0;
        thisPixel.g=0;
        thisPixel.b=0;
        theUnion.setRGB(x,y,thisPixel);
    }
}
```
Puzzle Time!

If you take this image and apply this code

```java
for(y=0;y<theUnion.height;y++){
    for(x=0;x<theUnion.width;x++){
        thisPixel = theUnion.getRGB(x,y);
        thisPixel.r=0;
        thisPixel.g=0;
        thisPixel.b=0;
        theUnion.setRGB(x,y,thisPixel);
    }
}
```

What will you get?
Puzzle Time!

If you take this image and apply this code

```java
for(y=0;y<theUnion.height;y++)
    for(x=0;x<theUnion.width;x++){
        thisPixel = theUnion.getRGB(x,y);
        thisPixel.r=0;
        thisPixel.g=0;
        thisPixel.b=0;
        theUnion.setRGB(x,y,thisPixel);
    }
```

What will you get?
Puzzle Time!

If you take this image and apply this code

```java
for(y=0;y<theUnion.height;y++){
    for(x=0;x<theUnion.width;x++){
        thisPixel = theUnion.getRGB(x,y);
        thisPixel.r=0;
        thisPixel.g=0;
        thisPixel.b=0;
        theUnion.setRGB(x,y,thisPixel);
    }
}
```

What will you get?
A Simple Cat Object
A Simple Cat Object
A Simple Cat Object

Properties
name  color  size  dob
A Simple Cat Object

Properties
name
color
size
dob

Methods
purr()
sayMeow()
begForFood()
shed()
The Takeaway

<table>
<thead>
<tr>
<th>What is a/an</th>
<th>How do you</th>
</tr>
</thead>
<tbody>
<tr>
<td>object</td>
<td>build a nested loop</td>
</tr>
<tr>
<td>dot operator</td>
<td>store strings</td>
</tr>
<tr>
<td>Pixel</td>
<td>iterate over all pixels in an image</td>
</tr>
<tr>
<td>property</td>
<td>access an object's properties</td>
</tr>
<tr>
<td>method</td>
<td>access an object's methods</td>
</tr>
<tr>
<td>ASCII table</td>
<td>store images</td>
</tr>
<tr>
<td>RGB</td>
<td></td>
</tr>
<tr>
<td>HSL</td>
<td></td>
</tr>
</tbody>
</table>

Homework: Look up what this is!