Variables

Review of Variables
User-defined Variables and State
Binary Numbers and Numerical Representation

Chapter 4, Examples 4-1, 4-2, 4-3, 4-4, Robot 2 (p. 54-56).
Chapter 8, Examples 8-1, 8-2, 8-3, 8-4, 8-5, 8-7, 8-8, 8-9 8-10, 8-11.
What does this code draw?

```javascript
let w = 50;

function setup() {
  createCanvas(100, 50);
  strokeWeight(20);
  w = 0;
}

function draw() {
  background(200);
  w = 100;
  point(w, height / 2);
}
```
How to MAKE A FLIPBOOK (1:51 - 7:39)
https://www.youtube.com/watch?v=Un-BdBSOGKY
Review of Variables (So Far)

- Definition of a variable
- Variable types
- Variable Declaration vs. Variable Assignment
- let, const, and var
- JavaScript Data types
- Mathematical operators
- Functions that Return a Value
Review of Variables (So Far)

- Definition of a variable

https://editor.p5js.org/sanghosuh/sketches/QjUjKTf4Q
https://editor.p5js.org/sanghosuh/sketches/splu9fRHx
Review of Variables (So Far)

- Variable types

1. Built-in
   - e.g.

2. Constant
   - e.g.

3. User-defined
   - e.g.
Review of Variables (So Far)

- Variable Declaration vs. Assignment vs. Initialization

```javascript
// Day 0
let piggyBank;

// Day n (n = 1, 2, ...)
piggyBank = 1;

// Day 0
let piggyBank = 1;
```
Review of Variables (So Far)

- let, const, and var

  - **let** are for "?" variables
  - **const** are for "?" variables
  - **var**?
Review of Variables (So Far)

- JavaScript Data types

  - **Number**
    - e.g., -5, 1, 0.12, -10.1

  - **Boolean**
    - e.g., true, false

  - **String**
    - e.g., “hello”, “123”

  - **Object**
    - e.g., an Array
Review of Variables (So Far)

- Mathematical operators

Order of operations:

```
B E DM AS
```

- Brackets
- Divide
- Multiply
- Add
- Subtract
- modulo
Review of Variables (So Far)

- Functions that Return a Value  
  e.g.,

\[
x = \text{random}(100);
\]

\[
y = \text{min}(50, \text{mouseY});
\]

\[
\text{strokeWeight}(\text{min}(50, 25));
\]
Review: Why User-defined Variables?

- To reuse the same value in your program
  - e.g. draw many squares of same size
- To make it easy to change a value later
  - e.g. make squares larger
- To make your code easier to understand
  - e.g. easier to read since "squareSize" means more than "10"

Starter: [https://editor.p5js.org/san.../l2M5yH89w]
Review: Why User-defined Variables?

- To **reuse the same value** in your program
  - *e.g. draw many squares of same size*
- To make it **easy to change a value later**
  - *e.g. make squares larger*
- To make your code **easier to understand**
  - *e.g. easier to read since "squareSize" means more than "10"*

Starter: [https://editor.p5js.org/sanghosuh/sketches/JjgKxQa63](https://editor.p5js.org/sanghosuh/sketches/JjgKxQa63)
[https://editor.p5js.org/sanghosuh/sketches/l2M5yH89w](https://editor.p5js.org/sanghosuh/sketches/l2M5yH89w)
We also need them...

- To **reuse the same value** in your program
  - e.g. *draw many squares of same size*
- To make it **easy to change a value later**
  - e.g. *make squares larger*
- To make your code **easier to understand**
  - e.g. *easier to read since "squareSize" means more than "10"*
- To **keep track of** something that changes
  - e.g. *object position, number of clicks*

Assignment 3

https://youtu.be/zTxu3dl5WNI?t=33
We also need them...

- **To reuse the same value** in your program
  - *e.g. draw many squares of same size*

- **To make it easy to change a value later**
  - *e.g. make squares larger*

- **To make your code easier to understand**
  - *e.g. easier to read since "squareSize" means more than "10"*

- **To keep track of** something that changes
  - *e.g. object position, number of clicks*

Assignment 3

https://youtu.be/zTxu3dl5WNI?t=33
Program State (cf. Lecture 04 Attributes)

- Current settings of all drawing attributes (e.g. fill, stroke, strokeWeight, etc.)
- What line of the program is executed next? (i.e. Sequential Control Flow)
- Current values of all variables (use variables to “save” and “update” state of things)
Using Variables to “Save State”

- Example: What kinds of state are there in a game?
animation

adding a bit to a variable each frame

assigning variable when mouse button event

Starter:: https://editor.p5js.org/cs105/sketches/0a9KXgYLo

https://editor.p5js.org/cs105/sketches/Mc|4bsUN|
Trace of variables

- see: “07 Variables (trace).pdf”
animation + print()

print() helps you understand the behavior

Starter: https://editor.p5js.org/cs105/sketches/0a9KXgYLo

https://editor.p5js.org/cs105/sketches/McJ4bsUNI
use mouse and key presses to control circle sizes, all using user-defined variables

function draw() {
    ...
    ellipse(width / 2, height / 2, redSize, redSize);
    ...
}

function keyPressed() {
    redSize = redSize + 3;
}
What does this code print on the 5th frame?

A. 1  
B. 10  
C. 11  
D. 14  
E. 15

```javascript
let a = 10;
let b = 0;

function setup() {
  createCanvas(100, 100);
}

function draw() {
  b = a;
  b = b + 1;
  print(b);
}
```
How Assignment (=) Works with Variables

A single equals sign = means: “assign calculated value on the right to the variable on the left”

- Only values of variables on right side are used.
- No variables are “linked.”
- Two variables do not share a location in memory.

```
let a = 10;
let b = a;
b = b + 1;
```

```
let a = 10;
let b = a;
b = b + 1;
```

```
let a = 10;
let b = a;
b = b + 1;
```
Tips for Understanding Execution Steps

- Add output (use print)
- Slow down code execution (change frameRate)
- Display current execution step (show frameCount)
Tip: Add output (use print)

- Print value of a variable to console
  ```javascript
  print("x: ", x); // prints "x: 123" (if x is 123)
  ```

- Print a message when function is called
  ```javascript
  function mousePressed() {
    print("mouse is pressed");
  }
  ```

- Use print to print a message at different places
  ```javascript
  print("HERE");
  print("draw start");
  ```

https://editor.p5js.org/sanghosuh/sketches/V5QMCgjM6
https://editor.p5js.org/sanghosuh/sketches/xvhVrVBNP
Tip: Slow down code execution & Display current step

- Make your program run artificially slow to see what happens on individual frames

```javascript
function setup() {
  ...
  // draw runs at only one frame-per-second
  frameRate(1);
}

function draw() {
  print("frame #" + frameCount);
}
```

https://editor.p5js.org/sanghosuh/sketches/4jDAYjtpv
Numerical Representation

- How are numbers represented in computers?
What are some ways we can represent numbers?

What are some ways we can represent numbers?

International Morse Code

1. The length of a dot is one unit.
2. A dash is three units.
3. The space between parts of the same letter is one unit.
4. The space between letters is three units.
5. The space between words is seven units.

Morse code depends on **dot** or **dash**
Numerical Representation

- How are numbers represented in computers?
Numerical Representation

...
### Numerical Representation

- **Binary Numbers**: 1 (on) and 0 (off)

  (Base-2 system)

<table>
<thead>
<tr>
<th>1</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>...</th>
<th>...</th>
</tr>
</thead>
</table>

| on | off | off | on | off | off | on | ... | ... |

![Intel Core i7 Chip Image](imageurl)
Numerical Representation

- How numbers are represented in a computer
  - Bits and Bytes

1 byte
(= 8 bits)
Numerical Representation

- How numbers are represented in a computer
  - Bits and Bytes

<table>
<thead>
<tr>
<th>Bin</th>
<th>Char</th>
</tr>
</thead>
<tbody>
<tr>
<td>0011 0000</td>
<td>0</td>
</tr>
<tr>
<td>0011 0001</td>
<td>1</td>
</tr>
<tr>
<td>0011 0010</td>
<td>2</td>
</tr>
<tr>
<td>0011 0011</td>
<td>3</td>
</tr>
<tr>
<td>0011 0100</td>
<td>4</td>
</tr>
<tr>
<td>0011 0101</td>
<td>5</td>
</tr>
<tr>
<td>0011 0110</td>
<td>6</td>
</tr>
<tr>
<td>0011 0111</td>
<td>7</td>
</tr>
<tr>
<td>0011 1000</td>
<td>8</td>
</tr>
<tr>
<td>0011 1001</td>
<td>9</td>
</tr>
<tr>
<td>0011 1010</td>
<td>:</td>
</tr>
</tbody>
</table>
Numerical Representation

- Limits of storage:
  - numeric overflow
  - precision

32- or 64-bit CPU

```
0 0 ... 0 0 0 0 0 0 0 0
```

32 or 64 bits
Numerical Limits

- There are limits to what numbers can be stored in a computer.

Numeric Overflow
https://editor.p5js.org/cs105/sketches/iPADgbyJh

let a = 9007199254740982;

function setup() {
  frameRate(2);
}

function draw() {
  a = a + 1;
  // watch the output!
  print(a);
}

Precision Errors
https://editor.p5js.org/cs105/sketches/MJevdLhI3

let a = 1;

function setup() {
  frameRate(1);
}

function draw() {
  a = a + 0.1;
  // watch the output!
  print(a);
}
let z = 200;

function setup() {
createCanvas(100, 50);
strokeWeight(20);
}

function draw() {
background(200);
z = min(z, 50);
point(z, height / 2);
}
What is this binary number in decimal?

A: 101
B: 8
C: 5
D: 4
E: 3

0101