

COMP 1010- Summer 2015 (A01)

Jim (James) Young

young@cs.umanitoba.ca

jimyoung.ca

Hello!

James (Jim) Young

young@cs.umanitoba.ca

jimyoung.ca

office hours T / Th: 17:00 – 18:00

EITC-E2-582

(or by appointment, arrange by email)

Text in processing: strings

Strings

(strings of characters.....?)

“hello”

“this is a pen!”

“Hello, my name is Jim!!”

Put some text inside quote marks

```
println(“hello world”);
```

look a little closer...

```
println("hello world");
```

our **parameter** is in double quotes ""

warning: this is not two ' or ` (single quotes)

hold SHIFT and press single quote ' by the enter key

"" is used to denote that we are providing text
what happens if we remove the ""?

compile-time error!!

Strings are strange: a new kind of data

So far, all data are “primitives”

simple nuggets of data

Strings are more complicated, larger chunks of data, different sizes, etc.

Strings are **Objects**. They act differently in many ways from your other data ☹️

the String data type

```
String variableName;
```

Notice the capital S! → object

```
String s = "Hello World!!";
```

```
println(s);
```

The EMPTY String! (a little zen?)

Question: what would be the shortest String that you could think of?

1 character long?

how about 0 characters long?

in Processing, the **empty string** is a string that is 0 characters long

you do it by: "" putting two quotes together

```
String nickName = "";
```


Combining two strings?

```
String firstName = "Jim";
```

```
String lastName = "Young";
```

```
String fullName = "Jim Young";
```

Wouldn't it be nice if we could simply build **fullName** from the other two?

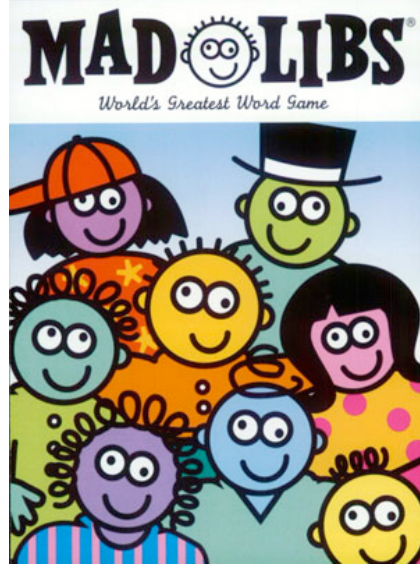
In Processing, we use + sign to **concatenate** strings:

```
fullName = "Jim" + "Young";
```

```
fullName = firstName + lastName;
```

```
fullName = firstName + " " + lastName;
```





madlibs

“<exclamation>! He said <adverb> as he jumped into his convertible <noun> and drove off with his <adjective> partner.”

Do this in Processing?

- a) Create variables for each of the unknowns
- b) Store answers for each
- c) Calculate the output String
- d) Display the output String

Graphical text in processing

Two new commands:

```
text(string, x, y); // draw string at x,y
```

```
textSize(size in pixels);
```

Some string

Coordinates point at
bottom left corner



What number is this?

五

いつつ

오

다섯

|||||

V

5

The number 5 can be represented in thousands of ways... that is text versus the abstract number

String type versus number types

is String “5” different than integer 5?

note: String is text, language-dependent representation, and numbers are abstract numbers independent of representation:

e.g., “1234.56” and “1,234.56” and “12,34.56” are string representations, and all are equivalent to the abstract number 1234.56.

```
String s = "5";
```

```
int i = s;
```

```
int i = 5;
```

```
String s = i;
```

So how to convert between a string and a number?

`int int(String)`

`float float(String)`

`String str(numerical)`

int takes a **String** parameter, and **returns** an **Integer** that you can use.

float takes a **String** parameter, and **returns** a **float** that you can use.

str takes a **numerical** parameter, and **returns** a **String** that you can use.

Fail cases?

```
String s = "1,234";
```

```
int i = int(s);
```

```
float f = float(s);
```

```
println(i);
```

```
println(f);
```


special case: string concat +=

```
String s = "output: ";
```

```
s += "other string"; // s = s + "other string"
```

```
s += 20;
```

```
s = s + 20; // converts 20 to "20"
```

Shortcut for number -> String

If you try to concatenate a String with a number, processing inserts the `str()` conversion for you

```
String s = "my age: ";
```

```
s = s + 19;
```

Example: Calendar

Setup globals

CAL_TOP (50)

CAL_LEFT (50)

CAL_DAYS (31)

CAL_SPACE (60)

TEXT_SIZE (30)

Draw header row.

S	M	T	W	R	F	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

Draw calendar numbers

Single for loop through days

Day -> row, column mapping?

Integer arithmetic

S	M	T	W	R	F	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

day	d/7	d%7		d/7	d%7		d/7	d%7
1	0	1	11	1	4	21	3	0
2	0	2	12	1	5	22	3	1
3	0	3	13	1	6	23	3	2
4	0	4	14	2	0	24	3	3
5	0	5	15	2	1	25	3	4
6	0	6	16	2	2	26	3	5
7	1	0	17	2	3	27	3	6
8	1	1	18	2	4	28	4	0
9	1	2	19	2	5	29	4	1
10	1	3	20	2	6	30	4	2

Draw text at each row, col.

Highlight selected cell

If a cell is clicked on, it gets selected and stays selected

But how to keep track?

Highlight selected cell

Global variable – selected!

Default day?

set to impossible

When drawing days, test if selected is the day

if so, highlight

How to select with mouse?

At each calendar spot, calculate left, right, top, bottom

Check if the mouse is within that, AND, the button is clicked

Debug!!! Header disappears if you click last day

```
final int CANVAS_SIZE = 500;
final int CAL_TOP = 50;
final int CAL_LEFT = 50;
final int CAL_DAYS = 31;
final int CAL_SPACE = 60;
final int TEXT_SIZE = 30;
int selected = 10;

void setup()
{
  size(CANVAS_SIZE, CANVAS_SIZE);
  textSize(TEXT_SIZE);
}

void draw()
{
  background(0);

  int x = CAL_LEFT;
  int y = CAL_TOP;
  String s = "S";
  text(s, x, y);

  x += CAL_SPACE;
  s = "M";
  text(s, x, y);

  x += CAL_SPACE;
  s = "T";
  text(s, x, y);

  x += CAL_SPACE;
  s = "W";
  text(s, x, y);

  x += CAL_SPACE;
  s = "R";
  text(s, x, y);

  x += CAL_SPACE;
  s = "F";
  text(s, x, y);

  x += CAL_SPACE;
  s = "S";
  text(s, x, y);

  // draw the numbers
  for (int i = 1; i <= CAL_DAYS; i++)
  {
    int row = i/7+1;
    int col = i%7;
    x = col*CAL_SPACE+CAL_LEFT; // left
    y = row*CAL_SPACE+CAL_TOP; // bottom
    int top = y-CAL_SPACE;
    int right = x+CAL_SPACE;

    if (mouseX>x && mouseX < right &&
        mouseY>top && mouseY < y && mousePressed)
    {
      selected = i;
    }

    if (selected == i) // current day is sel
    {
      fill(255);
      rect(x, y-CAL_SPACE, CAL_SPACE, CAL_SPACE);
      fill(0);
    } else { // not selected
      fill(255);
    }
    text(i, x, y);
  }
}
```