Inspiring innovations that change the world.

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

www.qatar.cmu.edu

SCPP CS Project

Lecture 5

Review
Quiz
Functions
Control structures
Activity: Morse Finch

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

1. Declare a variable of type double named temp and initialize it to 24.3.
2. Why is the setup() function special?
3. What gets printed when you run this?

```
void setup() {
  sayHello();
  sayGoodbye();
  sayHello();
}
void sayHello() {
  println("hello");
}
void sayGoodbye() {
  println("goodbye");
}
```

4. a) What gets printed when you run this?
   b) What is the return type of multiplyNumbers?
   c) Is the code in red a function call or a function definition?
   d) Is the code in blue a function call or a function definition?

```
void setup() {
  int first = 4;
  int second = 6;
  int total = 0;
  total = multiplyNumbers(first, second);
  println(total);
}
int multiplyNumbers(int x, int y) {
  int result;
  result = x * y;
  return result;
}
```

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Recap:
- Main function (setup)
- Function definition
- Calling the function
- Function return type
- Order of operations

```
void setup() {
  lightUpNose();
}

void lightUpNose() {
  Finch abe = new Finch();
  abe.setLED(255,0,0);
}
```

What about a return type other than void?

```
void setup() {
  int x;
  int y;
  int numStudents;


  x = numGirls();
  y = numBoys();
  numStudents = x + y;
  println(numStudents);
}


int numGirls() {
  return 4;
}


int numBoys() {
  return 9;
}
```

What if we want our function to have input?
   – Parameters

```
void setup() {
  int x = 4;
  int y = 6;
  int sum;


  sum = addNumbers( x, y);
  println(sum);
}


int addNumbers( int firstNum, int secondNum) {
  int total;
  total = firstNum + secondNum;
  return total;
}
```

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Why can't we use the variables x & y
from setup in the addNumbers function?

```
void setup() {
  int x = 4;
  int y = 6;
  int sum;


      sum = addNumbers( x, y);
      println(sum);
}


int addNumbers( int firstNum, int secondNum) {
  int total;
  total = firstNum + secondNum;
  return total;
}
```

What about a return type other than void?

```
void setup() {
  int x;
  int y;
  int numStudents;

  x = numGirls();
  y = numBoys();
  numStudents = x + y;
  println(numStudents);
}

int numGirls() {
  return 4;
}

int numBoys() {
  return 9;
}
```

What if we want our function to have input?
  – Parameters

```
void setup() {
  int x = 4;
  int y = 6;
  int sum;


  sum = addNumbers( x, y);
  println(sum);
}


int addNumbers( int firstNum, int secondNum) {
  int total;
  total = firstNum + secondNum;
  return total;
}
```

Carnegie Mellon Qatar
جامعة كارنيجي ميلون في قطر

Why can't we use the variables x & y
from setup in the addNumbers function?

```
void setup() {
  int x = 4;
  int y = 6;
  int sum;

      sum = addNumbers( x, y);
      println(sum);
}


int addNumbers( int firstNum, int secondNum) {
  int total;
  total = firstNum + secondNum;
  return total;
}
```

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

What if I really, really want a variable to exist for all the functions? I don't want to pass it as a parameter.

My friend, global variables are for you. This is where you declare a variable outside of **every** function.

Example:

```
int sum;  // Whee! Everyone knows who I am!

void setup() {
  int x = 2;
  int y = 3;
  sum = 0; // We still initialize it inside setup
  addNumbers(x,y);
  println(sum);
}


void addNumbers( int firstNum, int secondNum ) {
  sum = firstNum + secondNum;
}
```

Why can addNumbers have a void return type??

```
int sum;  // Whee! Everyone knows who I am!

void setup() {
  int x = 2;
  int y = 3;
  sum = 0; // We still initialize it inside setup
  addNumbers(x,y);
  println(sum);
}

void addNumbers( int firstNum, int secondNum ) {
  sum = firstNum + secondNum;
}
```

Carnegie Mellon Qatar

Why doesn't this work?

```
void setup() {
  int x;
  x = 4;
}

void printNum() {
  println(x);
}
```

We can fix it by making x a global variable

```
int x;

void setup() {
  x = 4;
  printNum();
}


void printNum() {
  println(x);
}
```

Carnegie Mellon Qatar