

ITEC 120

Lecture 5
Functions

Review

- Questions?
- Conditionals
 - If/else if / else
 - And/or/not

Functions

Objectives

- Learn about functions
- Syntax
- File structure

Functions

Visualization

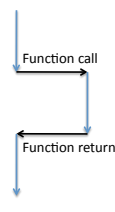
Current view of a program

```

0 - int x;
1 - int y;
2 - x=3;
3 - y=x*2;
4 - System.out.println(y);

```

New view



Functions

Currently

```

public class Example
{
    public static void main(String[] args)
    {
        int x;
        Scanner scan = new Scanner(System.in);
        x = scan.nextInt();
        System.out.println(x);
    }
}

```

Start 0
1
2
End 3

Functions

Reality

```

public class Example
{
    public static void main(String[] args)
    {
        int x;
        int x;
        Scanner scan = new Scanner(System.in);
        x = scan.nextInt();
        System.out.println(x);
    }
}

```

Start 0
1
2
End 6

Block of Code 1
Block of Code 2

Functions

Reality

```

0 int x;
1 Basics io = new Basics();
2 x = io.readInt();
3 Scanner scan = new Scanner(System.in);
4 int a = scan.nextInt();
5 return a;
6 io.println(x);

```

Block of Code 1

Functions

Rationale

- Accomplish X task
- Given X input, produce Y output
- Two parts
 - Function body (block of code that doesn't execute on its own)
 - Function call (calls the block of code to execute)

Functions

Functions

- Grouping mechanism

```
Scanner scan = new Scanner();
int num = scan.nextInt();
return num;
```

Put into FunctionContainer.java

```
int x;
FunctionContainer fc =
    new FunctionContainer();
x = fc.readInt();
```

- Two parts

- Function body
- Function call

Functions

Function Syntax

- [Access] [Return] [Name]() { }
- public int readInt() { }
- Access will be public for now
- Return is a primitive type (or void for none)
- Name is just like a variable name
- (begins parameters
-) ends parameters
- { begins code
- } ends code

Functions

Example

- Print a message

```
public void Print()
{
    System.out.println("Hello World");
}
```

- Square a number

```
public int readInt()
{
    Scanner scan = new Scanner(System.in);
    int num = scan.nextInt();
    return num*num;
}
```

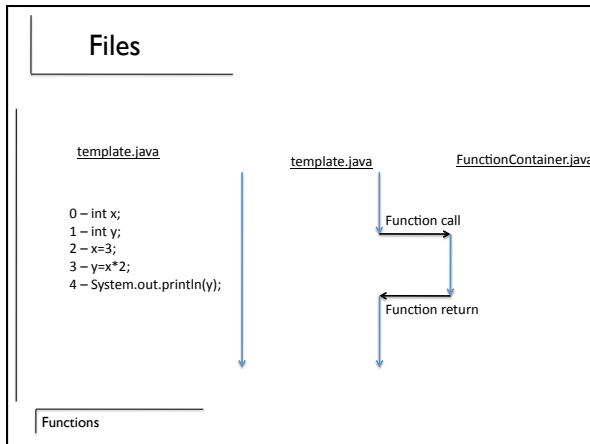
Functions

Example

- Square a number

```
public void Square(int x)
{
    System.out.println(x*x);
}
```

Functions



Function Container

```

import java.util.*;
public class FunctionContainer
{
    public void print()
    {
        System.out.println("Hello World");
    }
    public int readInt()
    {
        int num;
        Scanner scan = new Scanner();
        num = scan.nextInt();
        return num;
    }
}

```

Functions

Calling

Template.java
-Create a new FunctionContainer
-Use that variable combined with the name to call the function

```

public class template
{
    public static void main(String[] args)
    {
        FunctionContainer fc = new FunctionContainer();
        fc.Print();
        int x = fc.readInt();
    }
}

```

Functions

Example

- Writing a function that reads an int, squares, and returns it [Math.pow(var,2)]

Functions

Review

- Functions
 - Rationale
 - Syntax
 - Example

Functions