Intro. to Java Programming

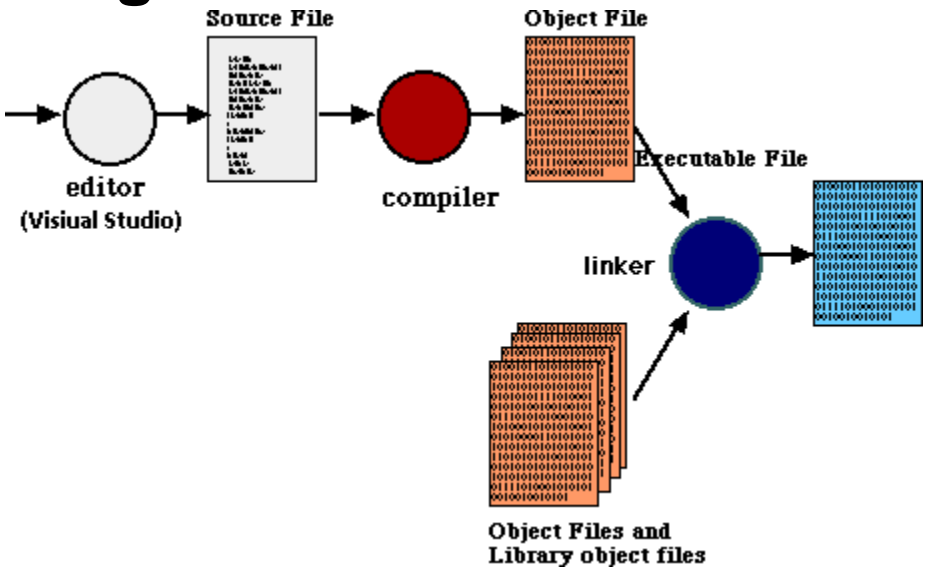
Programming - The process

- Understand the problem and requirements
- Design or select an algorithm to solve it
- Express (code) the algorithm in a programming language (e.g., C, Java)
- Test and debug the program until it works and meets the requirements

Hats I will wear this semester

You	Me
Programmer Debugger/User 	Programmer Debugger/User Professor 

Programming - The mechanics



Learning the setup of a program

- the setup is the same for many of your programs
- the setup must be in the SAME order as shown as below

Setup Example	
`// Prof. Lupoli` `// First Program - Learning the Setup`	Place name here Place what the program is going to be done
`// imports` `import java.util.Scanner;`	Place all imports here

`// class header`
`public class helloWorld`
`{`
	Create class header here

`// Set up Scanner` `static Scanner sc = new Scanner(System.in);`	Setup Scanner here

`// main()`
`public static void main(String[] args)`
`{`
 	
`// code goes here`
`System.out.println("hello world");`
 	
`System.out.println(Integer.MIN_VALUE);`
`System.out.println(Integer.MAX_VALUE);`
`System.out.println(Short.MIN_VALUE);`
`System.out.println(Short.MAX_VALUE);`
 	
`}`
`}`
	Create main here
	
	Where most of your code goes!!

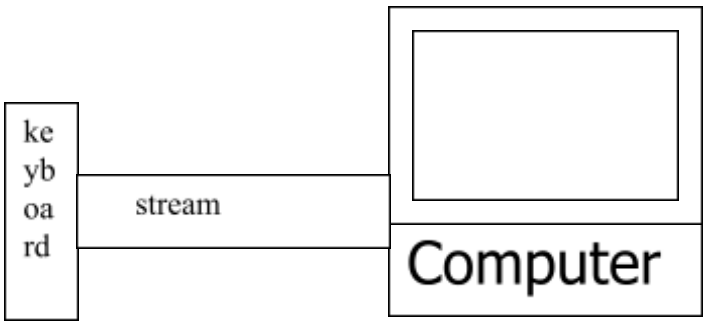
Description of Setup	
Import	Some commands and features require other "packages or libraries"
Class	Every program needs a class name, think of it as a program name. The name must be one word. (Or combined with a _)
Main	where most of your code will be placed. Starts from top to bottom.

Data types

Type	Examples/Definitions
int	**int yards = 202;**
	a WHOLE number, ranging from -2147483647 to 2147483647
short	**short coordinate = -12;**
	a WHOLE number, ranging from -32767 to 32767
long	**long debit = 10000000000;**
	a WHOLE number, ranging from -2^63 to 2^63-1
float	**float GPA = 3.99;**
	a real number in a floating point representation
double	**double mole = 1.2336483;**
	a double-precision real number in a floating point representation
long double	**long double weight = 7.7493343658792749;**
	a extended-precision real number in a floating point representation
char	**char choice = 'x';**
	ONE character, or a small INTEGER in the range from -128 to 127
boolean	**bool found = true;**
	a Boolean value can either AND ONLY be true(1) or false(0)
String	**String name = "Mr. Lupoli"; // yes a capital S**
	Holds a number of character together

Input and Output

I/O stream



sc.nextDouble( )
sc.nextInt( )

System.out.println( )
System.out.print( )

Printing a Line of Text

Print Example

```
// Mr. Lupoli
// 1st program

public class Hello
{
      public static void main(String args[])
      {
         System.out.print("Hello Class, ");
         System.out.println("I am Mr. Lupoli"); // what is the difference between
         System.out.println("We will learn JAVA!!"); // println and print??
      }

}
```

Draw a sizeable square on your paper. If it was a monitor what would it look like after the code above completed.

Use the code above to display YOUR name on line ONE, and your town and state on line TWO

Using System.out.println( ) with variables

```
int x = 0;          // MUST DECLARE ALL VARIABLE BEFORE USING
int y = 8;
```

Match the output	
(A) System.out.println("X is: " + x + " and Y is: " + y);	(1 or 2)
(B) System.out.println("X is: + x + and Y is: + y");	(1 or 2)

Options:

(1) X is + x + and Y is + y

(2) X is 0 and Y is 8

literal escape constants/command constants

JAVA Escape Sequences	
Escape Sequence	**Description**
\t	tab
\r	carriage return, go to beg. of next line
\\	backslash
\”	double quote
\’	single quote
\n	new line
\b	back space
\f	form feed

What will these statements below display??

```
System.out.println(“Hi Class!”);
System.out.println(“Good Luck!!! \n”);
System.out.println(“\t \t You’ll need it!!!”);
```

Introduction to the Scanner Class

- Thanks to Mike McCoy and Jordan Clark
- ***How to gather INPUT from the keyboard***
- The scanner class is a STANDARDIZED class that uses different methods for READING in values from either the KEYBOARD or a FILE.
- Must import
  - java.util.Scanner;
- Must “start” the Scanner (look at setup)

Intro. To Methods in Scanner class

- Just remember to first
  - IDENTIFY what exactly you wish to read in (or get from the user!!)
  - HOW you want to use it.
- Remember a numeric value CAN be read in as a String!!
- Methods in the class are broken down into two categories
  - next() (reads value)

String	next() Finds and returns the next complete token from this scanner.

```
// to read in a SINGLE char

char letter;
letter = sc.next().charAt(0);

String name;
name = sc.next();
sc.reset(); // use after next since it might look for more
```

double	nextDouble() Scans the next token of the input as a double.

```
double price;
price = sc.nextDouble();
```

Proper setup for input

- Remember, the user is probably not smart
- Help them and yourself out

Setup for input

Bad

```
import java.util.Scanner;

public class HelloWorld {

	static Scanner sc = new Scanner(System.in);

	public static void main(String[] args)
	{
		System.out.println("hello all");


		int integerValue = sc.nextInt();

	}

}
```

```
hello all
```

Good

```
import java.util.Scanner;

public class HelloWorld {

	static Scanner sc = new Scanner(System.in);

	public static void main(String[] args)
	{
		System.out.println("hello all");

		// tell user what they need to input
		System.out.println("Please enter an integer value");

		// grab input and store in a variable
		int integerValue = sc.nextInt();

		// confirm input by displaying variable
		System.out.println("you entered " + integerValue);

	}

}
```

```
hello all
Please enter an integer value
23
you entered23
```

What is a token??

See if you can figure it out from these examples??

	Token count
Lupoli	1
98	1
Prof. Lupoli!	2
123.012	1
Lupoli needs a vacation	4
! ! !	3

Rest of Scanner Methods

float	nextFloat() Scans the next token of the input as a float.

```
float amount;
amount = sc.nextFloat();
```

int	nextInt() Scans the next token of the input as an int.

```
int score;
score = sc.nextInt();
```

String	nextLine() Advances this scanner past the current line and returns the input that was skipped. THIS IGNORES SPACES WITHIN A USER INPUT!!!!

```
String entireName;
entireName = sc.nextLine();
```

Complete the exercise below

Which Scanner method would you use?

	Input is of what Datatype?	Scanner method needed
Lupoli		
98		
Prof. Lupoli!		
123.012		
Lupoli needs a vacation		
! ! !		

REMEMBER WE HAVE NO IDEA WHAT VALUE THE USER WILL ENTER!!!

First Scanner Example

```
// Prf. Lupoli
// Tests inputs
import java.util.Scanner;  // must import for Scanner usage

public class firstScan
{

      static Scanner sc = new Scanner(System.in); // start Scanner

      public static void main(String[] args)
      {
            int age = -1; // set a DEFAULT value

            System.out.println("How old are you?");
            age = sc.nextInt(); // grab value from keyboard (user)

            int dogAge = age * 7;

            System.out.println("You are " + dogAge + " years old in DOG YEARS");
      }
}
```

1. Identify where the import statement is located
2. Identify where the scanner command is located
3. Identify type of data (float, String, int) is being read in
4. Identify where the output is taking place

Inputting a Number	How old are you? 25 You are 175 years old in DOG YEARS
Inputting a Decimal	How old are you? 5.75 Exception in thread "main" java.util.InputMismatchException at java.util.Scanner.throwFor(Unknown Source) at java.util.Scanner.next(Unknown Source) at java.util.Scanner.nextInt(Unknown Source) at java.util.Scanner.nextInt(Unknown Source) at firstScan.main(firstScan.java:12)
Inputting a String	How old are you? Emily Exception in thread "main" java.util.InputMismatchException at java.util.Scanner.throwFor(Unknown Source) at java.util.Scanner.next(Unknown Source) at java.util.Scanner.nextInt(Unknown Source) at java.util.Scanner.nextInt(Unknown Source) at firstScan.main(firstScan.java:12)

Why did entering a decimal (float) or String break the program??

Input/Output Exercise

```
// First Program - Learning the Setup

import java.util.Scanner;

public class helloWorld
{
	static Scanner sc = new Scanner(System.in);

	public static void main(String[] args)
	{

		String name, address;

// 1. ) Create the CODE to LITERALLY display YOUR name, and address (No variables yet.)



// 2.) Create the CODE to ask AND ACCEPTS the user's name and address, USE THE VARIABLES
DECLARED FOR // YOU ALREADY!! Hint: Which scanner functions will you need?



// 3.) Create the code to display their name and address that THEY type in. NOT yours!!



	}
}
```

Use of Comments

//	/* … code … */

`// Mr. Lupoli`
`// Project 1`
`// 6/22/03`
`// Period 1`
 	
`//  "//"reserves REST of line for a`
`// comment`
 	
`// used for ONE line comments`
 	
`public static void main(String args[])`
`{`
`int counterValue;// sentinel value`
	`/*`
	`Mr. Lupoli`
	`Project 1`
	`6/23/03`
	`Period 1`
	`*/`
	
	`/*`
	
	`" /* " reserves whole block`
	`until you end it with a " */ "`
	
	`used for MULTIPLE lined comments`
	`*/`

Why use comments?

- For notes
  - to yourself
  - to me!!
- For commenting out unfinished lines of code
  - skipping unfinished functions
- To understand what the code is doing!!
- watch where you put them!!

Reserved Words

- case sensitive
- can not be used as variable or function names
- ex

`auto`	`default`	`enum`	`register`	`struct`
`break`		`extern`	`return`	`switch`
`case`	`do`	`float`	`short`	`typedef`
`char`	`double`	`for`	`signed`	`union`
`const`		`if`	`sizeof`	`unsigned`
`continue`	`else`	`int`	`static`	`void`
		`long`		`volatile`
				`while`

Coding Penmanship

- nested blocks
  - used for
    - compound statements
    - iteration (repeating loops of code)
    - conditional (if (x < 10)…)

Style and a COMPLETE example w/ Scanner

```
import java.util.Scanner;

class Example
{
	public static void main(String [] args)
	{
		Scanner sc = new Scanner(System.in);

		// variables
		char user;

		System.out.println("Will I do my work in Mr. Lupoli's Class??");

		// have user press "Y" or "N"
		user = sc.nextChar(); // reads what user typed

		if(user == 'Y') // will pass
		{ System.out.println("Then I will pass JAVA, and take JAVA AB next year!"); }
		// ONE LINE

		if(user == 'N') // will fail
		{
		System.out.println("Then I will not pass Mr. Lupoli's class.");
		System.out.println("And my parents will be upset!");
		// TWO LINES OR MORE
		}
	}
}
```