CS 149

Professor: Alvin Chao

Methods Review

1. Java programs are organized into classes, each of which has one or more methods, each of which has one or more statements. Writing methods allows you to break down a complex program into smaller blocks of reusable code.

2. Each statement a program **invokes** (or calls) a method. At the end of a method, Java **returns** to where it was invoked.

```
public static int abs(int a)
public static double log(double a)
public static double pow(double a, double b)
public static double random()
public static int subtractExact(int x, int y)
```

- Consider the following methods defined in the Math class
- value = abs(-5); *// Error cannont find symbol*
- value = Math.abs(-5); *//correct*
- The period in this example is called the dot operator. When reading the above code out loud, you would say "math dot abs".

Model 2 Math Methods

In the JavaDoc on the right for the Math Class

- What type of value does Math.random() return?
- When invoking a method, what do you need to specify before and after the method name?
- When defining a method, what do you need to specify before and after the method name?
- Define a method named average that requires two integers x, and y and returns a double. (This is called the **method signature**)
- How many parameters and arguments does each method have?

| Modifier and Type	Method and Description
static int	**abs**(int a) Returns the absolute value of an int value.
static double	**log**(double a) Returns the natural logarithm (base *e*) of a double value.
static double	**pow**(double a, double b) Returns the value of the first argument raised to the power of the second argument.
static double	**random**() Returns a double value with a positive sign, greater than or equal to 0.0 and less than 1.0.
static int	**subtractExact**(int x, int y) Returns the difference of the arguments, throwing an exception if the result overflows an int.

Math Methods contd.

Method	# Params	# Args
abs		
log		
pow		
random		
subtractExact		
println		

Modifier and Type	Method and Description
static int	**abs**(int a) Returns the absolute value of an int value.
static double	**log**(double a) Returns the natural logarithm (base *e*) of a double value.
static double	**pow**(double a, double b) Returns the value of the first argument raised to the power of the second argument.
static double	**random**() Returns a double value with a positive sign, greater than or equal to 0.0 and less than 1.0.
static int	**subtractExact**(int x, int y) Returns the difference of the arguments, throwing an exception if the result overflows an int.

Model 3 Stack Diagrams

How many methods does the program call?

How many variables does the program have?

How can two variables of the same name have different values?
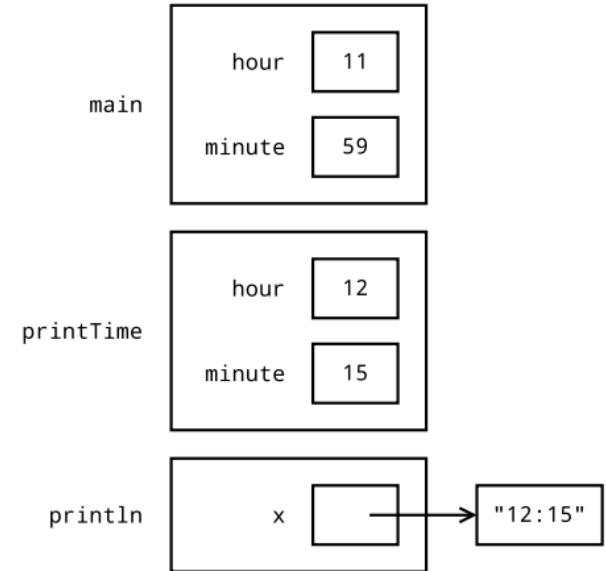

```
public static void printTime(int hour, int minute) {
    System.out.println(hour + ":" + minute);
}

public static void main(String[] args) {
    int hour = 11;
    int minute = 59;
    printTime(12, 15);
}
```

Stack Diagrams

1. Draw a stack diagram just before println is called. Assume the user inputs the value 10.

```java
public static void show(double c) {
    double f;
    String str;
    f = c * 1.8 + 32;
    str = String.format("%.1f C = %.1f F\n", c, f);
    System.out.println(str);
}

public static void main(String[] args) {
    double c;
    Scanner in = new Scanner(System.in);
    System.out.print("Enter temperature in Celsius: ");
    c = in.nextDouble();
    show(c);
}
```

- **Acknowledgements**

Parts of this activity are based on materials developed by Chris Mayfield and Nathan Sprague.

_____

</end>