

```
// Demo RPGGame.java
```

# Nested While Loops

```
while(condition1) {  
    while(condition2) {  
        ...  
    }  
    ...  
}
```

```
// this program will print every letter on its own
// line of whatever the user enters. Once the user
// enters quit, the program stops.
```

```
int counter;

System.out.println("Enter String");
String input = keyboard.nextLine();
while(!input.equals("quit")){
    counter = 0;
    while(counter < input.length()){
        System.out.println(input.charAt(counter));
        counter++;
    }
    System.out.println("Enter next string");
    input = keyboard.nextLine();
}

System.out.println("Program done.");
```

# Nested For Loops

```
for( init1; condition1; update1 ) {  
    for( inti2; condition2; update2 ) {  
        ...  
    }  
}
```

update1 does not execute until condition2 fails. Thus update1 does not execute until the entire life of the inner for loop has run its course. The inner for loops starts all over again once update1 executes.

```
for(int i = 0; i < 4; i++) {  
    for(int j = 0; j < 3; j++) {  
        System.out.println("i: " + i + ", j: " + j);  
    }  
}
```

Output:

```
i: 0, j: 0  
i: 0, j: 1  
i: 0, j: 2  
i: 1, j: 0  
i: 1, j: 1  
i: 1, j: 2  
i: 2, j: 0  
i: 2, j: 1  
i: 2, j: 2  
i: 3, j: 0  
i: 3, j: 1  
i: 3, j: 2
```

```
// Printing out a grid
System.out.println("How wide do you want this grid?");
int width = keyboard.nextInt();

System.out.println("How high do you want this grid?");
int height = keyboard.nextInt();

// print out grid. Outer is height, inner is width
for(int i = 0; i < height; i++) {
    for(int j = 0; j < width; j++) {
        System.out.print('*');
    }
    System.out.println();
}
```

```
// What is the value of counter when these loops  
// are done iterating?
```

```
int counter = 0;  
for(int i = 0; i < 5; i++){  
    for(int j = 0; j < 6; j++){  
        counter++;  
    }  
}
```

```
// Triangle of asterisks

System.out.println("How many levels?");
int levels = keyboard.nextInt();
for(int i = 1; i <= levels; i++){
    for(int j = 1; j <= i; j++){
        System.out.print("*");
    }
    System.out.println();
}
```



# Break/Continue

- Use `break` to exit the closest loop
- Use `continue` to jump to the next iteration of the loop

# Scope

- A variable only exists in the block in which it's defined. Blocks are denoted by { , }

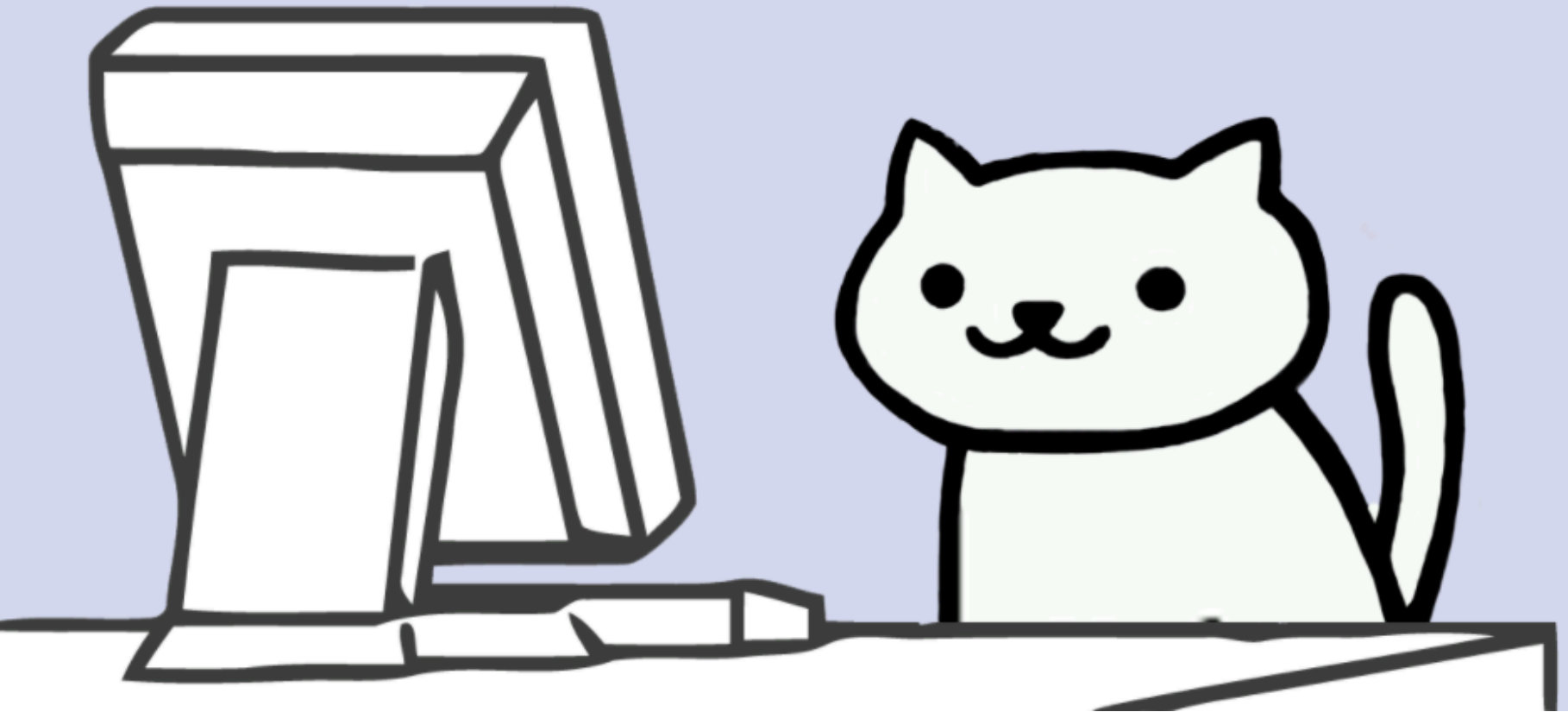
# Scope

```
int x = keyboard.nextInt();  
if(x == 0){  
    int y = 5;  
}  
System.out.println(y); // This is an error
```

# Scope

```
int x = keyboard.nextInt();  
int y;  
if(x == 0){  
    y = 5;  
}  
System.out.println(y); // This compiles
```

# Practice Problems



```
int size = keyboard.nextInt();
for (int i = 1; i <= size; i++) {
    for (int j = 1; j <= size; j++) {
        if (i == j)
            System.out.print('*');
        else
            System.out.print(' ');
    }
    System.out.println();
}
```

User enters 5 for input

A

```
*
 *
  *
   *
    *
```

B

```
      *
     *
    *
   *
  *
```

C

```
*
*
*
*
*
```

D

\*\*\*\*\*

```
int size = keyboard.nextInt();
for (int i = 1; i <= size; i++) {
    for (int j = 1; j <= size; j++) {
        if (i < j)
            System.out.print('*');
        else
            System.out.print(' ');
    }
    System.out.println();
}
```

User enters 5 for input

A

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

B

      \*  
      \*\*  
      \*\*\*  
      \*\*\*\*  
      \*\*\*\*\*

C

\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

D

      \*  
      \*\*  
      \*\*\*  
      \*\*\*\*  
      \*\*\*\*\*

```

int size = keyboard.nextInt();
for (int i = 1; i <= size; i++) {
    for (int j = 1; j <= size; j++) {
        if (i == 1 || i == size || j == 1 || j == size)
            System.out.print('*');
        else
            System.out.print(' ');
    }
    System.out.println();
}

```

User enters 5 for input

A

```

*****
*   *
*   *
*   *
*   *
*****

```

B

```

      *
      *
      *
      *
*****

```

C

```

*
*
*
*
*****

```

D

```

*
**
***
****
*****

```



```
int x = 5, y = 10, z = 20;  
if (x > y)  
    if (x < z)  
        x = 99;  
else  
    x = 100;  
System.out.println(z);
```

- A.) 5
- B.) 10
- C.) 20

```
Scanner in = new Scanner(System.in);
int x = in.nextInt();
int y = in.nextInt();
int z = in.nextInt();
if (x < y)
    if (y < z)
        System.out.println("yes");
else
    System.out.println("no");
```

A. Give a sequence of 3 numbers that results in the program above printing "yes". \_\_\_\_\_

B. Give a sequence of 3 numbers that results in the program above printing "no". \_\_\_\_\_

C. Give a sequence of 3 numbers that results in the program above printing nothing. \_\_\_\_\_

```
/* Complete the program below such that it calculates
the average length of a line given by the user. Use
the nextLine() method (belonging to the Scanner
object) to get the user's next line of input. Also
consider using the length() method belonging to
String. The program should stop once the user enters
"quit".
*/
```

```
Scanner keyboard = new Scanner(System.in);
```

```
int charCount = 0;
```

```
int lineCount = 0;
```

```
String userInput = keyboard.nextLine();
```

```
// Your code goes here
```