2nd Day:

Short circuiting:

A && B: If A is true, must go on to check B

A || B: If A is true, does not go on to check B but if A is false, must go on to check B if it is true or not

* Strings are objects in Java
* For string, must start with quotes and end with quotes
* \n (new line)
* The new line counts as one character in a string
* The tilda “~” is 7e in java

Char is a primitive type:

* Char is put in apostrophes
* Char is interchangeable with ints
* Can also cast an int to a char and can print the letter

Strings:

* “==” compares and sees if it is the same objects (basically compares memory addresses)
* .equals() checks if they have the same value
* .charAt
* .equals
* .indexOf
* .length
* .substring(4,7)
* .substring(7)

Nulls:

* Null can be assigned to objects not primitives
* Null pointer exception, since c is null, cannot call a method on it vs string “a” containing a string
* For checking nulls “!=” and “==” can be used

Errors:

* Syntax errors (compile-time errors)
* Runtime errors
* Semantic errors (Logic errors)
* Look through code and find the error
* Trace the code: Write values of variables and update each variable as it changes and trace where it went wrong
* Insert print statements to determine where a variable got messed up
* Run a debugger

Files:

* .next
* .nextLine
* output.append()
* The output must have output.close()
* Catch works if there isn’t no file, meaning it will print whatever it is meant to print to the console
* Try is what it will do if the file exists

For exam:

System.out.println()

Arrays:

* int [] intArray = new int [5];
* Creates an array of size 5
* For java, it fills the array with 0’s for int, 0.0 for doubles
* int [] intArray2 = intArray; // points to the same address in intArray
* intArray 2 points to the same places in intArray
* For arrays there are no parentheses for .length
* When 1d arrays are used in a method, it is a pass by reference meaning it changes the actual value within the array

For each loop:

* for(int x : intArray)
* Not good for modifying values
* Not good for accessing elements
* Good for not changing and implementing through the array