5 volt current = 1 bit binary

0 volt current = 0 bit binary

**Material Review**

* variables
  + declaration
  + giving value
  + byte
  + char
  + int – *integer* variable that can have numbers as value
  + float
  + boolean – variable that can have logical values true or false
  + String – variable that can store text as value
* operators
  + +
  + -
  + \*
  + /
  + %
  + **=** - assign values to variables
  + **+=** - add number on the right to the variable left
  + **-=** - subtract number on the right to the variable left
  + **\*=** - multiply number on the right to the variable left
  + **/=** - divide number on the right to the variable left
  + **==** - equal to
  + **!=** - not equal
  + **>=** - greater than / equal to
  + **<=** - less than / equal to
  + **>** - greater than
  + **<** - less than
  + **||** - OR operators, one statment has to be true to return true
  + **&&** - AND operator, both statment has to be true to return true
  + !

**! JAVA reads statments left to right, if they not separeted by brackets!**

boolean a = true || true && false || false && true

**IF / ELSE IF / ELSE statements:**

if (b)}

System.out.println(„true”)

{

else if (a) {

System.out.println („true”)

}

else {

System.out.println(„false”)

}

*if* – Checking the the condition given, if the condition is true execute statment, if false skips the code

*else if* **-** Checking the the condition given, if the condition is true execute statement, if false skips the

code

*else* – if every statments above is false (its does not have a condition), then it executes the else statment

**WHILE / DO WHILE / FOR (loop) statment:**

*Using WHILE loop:*

int a = 0;

while (a < 100) {

System.out.println(a);

a ++; (it means add 1 to variable every timet he statement runs)

}

System.out.println(„Loop finished!”)

It is going to run for 100 time, until it adds to a = 100, then it comes out of the while loop.

*Using FOR loop:*

for (int a = 0 ; a < 100; a++) {

System.out.println(a)

}

System.out.println(„Loop finished!”)

Give the statement a variable, then a coondition, then an operation.

*Using DO WHILE loop:*

int a = 10;

do {

System.out.println(„I print it out even the statement is false

} while (a < 10) {

System.out.println(„This loop is True!”)

}

System.out.println(„Loop finished!”)

The DO statement going to run the code given before the while loop checks its condition given, so we get an output even the loop condition is false and not going tor un.

* difference between variable declaration and statements
* **System.out.println()** – prints out the value givin on the Terminal
* Hello World
* escape character
* "Java's stuff" and 'stuff of Java'
* concatenation
* condition, if, else
* loops: for, while
* block
* scanner
  + System.in
  + nextLine()
  + nextInt()