//THE MAGIC INCANTATION
public static void main(String[] args) {

//LOOPS
//rule of thumb: if you know the iteration count use for, if not, use while.

// FOR LOOP
for (init; boolean expression; increment) {
statement 1;
statement 2;
}

//CLASS ANATOMY
public class Circle {
private double radius; //instance variable decl. (automaticially intitalised)

```
  public Circle(double newRadius) { //constructor (circle: constructor name)
      radius = newRadius; // instance variable
  }
  public double getArea() { //instance method
      return radius * radius * Math.PI;
  }
```

}

//ACCESS MODIFIERS
Modifier class package global
Public yes yes yes
Default yes yes no
Private yes no no

//ARRAYS
myarr[3] // elements acessed by bracket notation (rather than dots)
myarr.length //exception to the rule

java.unit.Arrays //dis yourhomeboy
arraycopy() //copies shit
copOfRange(array, from, to) //do not need to create a destination as returned by method
sort() //puts into accending order
binarySearch() //searches for value and returns its index
equals() //compares two arrays to see if equal
fill() //places a specific value at each index

//how to print arrays

//add more!

//EQUALS (a == b vs. a.equals(b))
//objects compare instances (position in memmory), primatives compare value
.equals //to stay on the safe size and compare states not instancea

//FORMATTING
source -> format //magically improves indentation & spacing
refactor (->) rename // repaces all instances of x with y

//FORMAT SPECIFIERS
String.format(string format, object … args)
System.out.format()

// Specifier Data Type Output
//
// %a floating p. hex output
// %b any "true" if !null "false" otherw.
// %c character unicode character
// %d integer decimal integer
// %e floating p decimal number in scientific notation
// %f floating p. decimal number
// %g floating p. fuck knows
// %h any hex string value (hash code)
// %n none platform specific line separator
// %o integer octal number
// %s any type string value
// %t date/time date/time conversion
// %x integer hex string

//needs work!

// SPECIAL CHARACTERS
\b //backspace
\t //tab
\n //new line
\f //from feed
\r //carriage return
\" //double quote "
\' //single quote '
\ //backslash

//CONDITIONAL OPERATORS
&& // AND
|| // OR
?: // if then else shorthand operator
someCondition ? value1 : value2
//(if) someCondition ? (is true) (assign the value) value1 : (else) value2;

//THIS
//point class
public class Point {
public int x = 0;
public int y = 0;

public Point(int x, int y) {
this.x = x;
this.y = y;
}

//ERRORS

throw new IllegalArgumentException("…")
Objects.requireNonNull(object)

DecimalFormat("####.##")

super //calles overridden method from superclass