Java

Function	Code	Comments
Declaring a class	public class Name {Content}	
Declaring a main method within a class	public static void main (String[] args) {Content}	
Modifiers	• Public void = can be accessed anywhere • private void = can only be accessed within its class • protected void = can only be accessed within package and its classes • void = can only be accessed within package	define where methods or variables can be used
Declaring a variable	ex.: int myVariable = 5;	• Java needs to know upfront what kind of data is stored in the variable and the name of it • If variable is declared within a method, it is local to that method (only exists within that method) • A global (class) variable needs to thus be declared before the main method
Declaring your own method	static void MethodName() {}	Important: declaring it outside of {} of main method, but calling it within {} of main method
Declaring method that takes additional input when calling it	static void MethodName (DataType VariableName) {}	The method works with VariableName (e.g. a String called name) When calling the method, a variable is passed on in the () (e.g. ("Charlie"))
Method that performs a calculation and returns a value	static int MethodName() {return Calculation of Value;}	returns the specified value when called
Executing code	;	Always needs to end with ";" (unless its comment)
Comments	// Content	//

Printing out	System.out.println();	important: return statements do not print, they need to be printed normally

Data types	double = float
	int = integer
	String = Text
	boolean = boolean
	JFrame = Object	

Concatenation	System.out.println(x + y);	
Conditions	if (Condition), else if, else, for(condition), while (condition) work with {} and booleans true and false	!= is not, == is && and, || or
Declaring objects	ObjectType ObjectName = new ObjectType();	e.g. for ObjectType is JFrame
Applying methods and properties to objects	ObjectName.method(properties);	Base of object-oriented programming
Referring to an object of a class within that class	this.Object	
Packages	Folder for java files. allows to import and use classes and methods from a package	
Declare constants	public static final DataType ConstantName = Constant;	can be used anywhere in program
Inherit class	class ChildClassName extends ParentClassName	child can use all the methods of parent
Referring to a method of a parent class within a child class	super.Method	
Receiving user input	Import Scanner class	
Use methods from a different class (interface)	Public class ClassName implements OtherClassName {}	
Cast from one numerical type to another numerical type	e.g. intValue = (int)floatValue	