INTRODUCTION

Object Oriented Programming uses classes to encapsulate data and the functions needed to manipulate that data. Objects are instantiated according to the class design. An advantage to OOP is reuse of classes.

IDE: Program Editor + Compiler + Run-Time Environment integrated via GUI.

- A Java program is first compiled into a Processor independent Byte code.
- Byte codes are interpreted at run time by JVM (JAVA Interpreter).
- JVM stimulates a virtual processor with its own instruction set, registers, and instruction pointer. Thus, to run Java program we need only JVM. (& they are available at every computing platform).
- Performing operations in order - Sequential Processing
- By convention, CONSTANT_IDENTIFIER consists of all capital letters, and embedded words are separated by an underscore. This makes it stand out in the code and easy to identify as constant. They are also defined at the top of the program where they can be seen easily. Assignment expression cannot include another variable unless that variable has been defined previously.

Identifiers are symbolic names for classes, methods, and data.
Rules:

- They always start with a letter
- They have no spaces
- They are case-sensitive.

The JAVA compiler follows a set of rules called *operator precedence* to determine the order in which the operations should be performed.

ALU performs division of two integers. It can calculate only an integer results. No rounding is performed. Any fractional part is truncated.

DIVISION BY ZERO yields:

4.3/0.0 [infinity]

0.0/0.0 [NaN - Not a Number]

4/0 [Error]

ERRORS:

Compiler Errors/Syntax Errors: A Non-proper, Not allowable sequence of characters or words given a particular language. {Incorrect JAVA Syntax or Misspellings}

- Undefined Variable Name
- Undefined Keyword
- missing {}, ; and malformed comments

Semantic Error/ Run-Time Errors: Detected by JVM. Due to incorrect use of classes or exceptions.

- Infinite Loop
- Output Incorrect
- Divide by 0 [error messages at runtime]
- Errors may be intermittent

Logic Errors due to incorrect program design or incorrect implementation of the design.

The arithmetic promotion of operands is called implicit type casting because the compiler performs the promotions automatically, without our specifying that the conversions should be made. Note that the data type of any promoted variable is not permanently changed; its type remains the same after the calculation has been performed.

8 Primitive Data Types:

- double (double precision type)
- float (single precision type)
- Boolean
- char (One Unicode character)
- int
- byte
- long
- short

User input can be read into program in several ways.

- From JAVA console (scanner class)
- From a dialog box
- From a file
- Through a GUI

The scanner class is defined in the java.util package.
import java.util.Scanner;

Instantiating a Scanner object named scan and associate System.in as the data source.
Scanner scan = new Scanner (System.in);

A Scanner object divides its input into sequences of characters called tokens, using delimiters. The default delimiters are the standard white-space.

If the user's input (next token) does not match the data type of the next... Method call, then an InputMismatchException is generated and the program stops.

Classes can also define static methods, which can be called without instantiating an object. Those are also called class methods. The API of these methods has the keyword static before the return type. It provides a quick, one time functionality without requiring the client to instantiate an object. Processor time and memory wastage is saved. Class, or static, methods are invoked using the class name, rather than an object reference.
ClassName.staticMethodName (argumentList);

The order in which the instructions are executed is called the flow of control of the program.
there are essentially four types of flow of control:

- Sequential Execution
- Method Calls
- Selection
- Looping

SELECTION:

- if

- if-else
- if-else if
- conditional operator(?:)
- switch

Equality Operators (they are binary operators):results in a Boolean value/primitive numeric/obj reference. Used to test whether two operands are equal.
==
!=
Relational Operators (they are binary operators): results in a primitive numeric type.
<
<=
>
>=
[Boolean]Logical Operators (they take Boolean expressions as operands):
! - inverts value of operand
&& both true, then returns true
|| both false, then returns false

Precedence: Unary NOT has highest precedence of the relational and logical operators, followed by relational operators, then equality operators, the AND, then OR.

The scope of a variable is the region within a program where the variable can be referenced, or used. When we declare a variable, its scope extends from the point at which it is declared until the end of the block in which we declared it.

The if/else/if statement is appropriate when the data falls into more than two mutually exclusive categories and the appropriate instructions to execute are different for each category.

‘if’ statements can be written as part of the true or false block of another if statement. These are called nested if statements. Typically, you nest if statements when more information is required beyond the results of the first if statement.

We always need to test the condition of the test plan as well in the test plan.

DANGLING ELSE: A common error is writing else clauses that don't match any if conditions. Compile error known as "else without if"

When if statements are nested, the compiler matches any else clause with the most previous if condition that doesn't already have an else clause. Because rounding errors can be introduced in floating point calculations, do not use the equality operators to compare two floating point numbers. Instead, compare the absolute value of the difference between the numbers to some threshold value.

Using the equality operator on objet references compares the values of the references, not the object data. Two object references will be equal only if they point to the same object.

Switch statement: evaluates an integer or character expression or a String. It then compares the expression's value to case constants. Executes until a break statement or end of switch block is encountered.