Program Requirements

# Overview

* Your program must compile and run
* Your program should solve the stated objectives of that program
* Your program should produce a valid output (invalid outputs are a worthless program)
* Your program should use the current techniques and concepts that we are covering

# Formatting

Your program must adhere to the Java standards and conventions: [Java’s naming conventions](https://www.oracle.com/java/technologies/javase/codeconventions-namingconventions.html)

Table

Description automatically generated

Your program must have proper indentations, spacing of blocks of code, and block styles

public static void main(String[] args) {

for (int i = 0; i < 10; i++) {

if (i % 2 == 0) {

System.out.println("Hello world");

}

}

}

Indentation makes code easier to read. It helps the reader see what code is nested where, and where structures like if statements and loops start and end.

You can have VS Code fix all your indentation for you! Press **SHIFT + ALT + F** and VS Code will automatically indent your code and add some structural annotations to your code.

# Comments

* Your program must have appropriate comments.
* At the beginning of your program (or the “test” program), you must have:
  + Name, date, and course
  + Files needed or produced if any
  + Describe what the program does, NOT how it does it.

// Student Name

// 1.2.2023

// CSCI 3302 Section 001

//

// Files: imdb.txt (input) output.txt (output)

//

// IMDB Search

// Searches a text file containing information about the top 250

// rated movies on IMDB for movies containing a particular word in the title,

// then prints the results to an output file.

* Each method must have a comment containing
  + Description: overview of what the method does
  + Parameters and Preconditions: information about the parameters and assumptions about them
  + Postconditions: conditions and expectations upon method completion

/\*\*

\* Rounds a given number to a given number of digits.

\* @param num - the double to be rounded

\* @param digits - positive integer for the number of digits to round to

\* @return the rounded number as a double

\*/

public static double roundN(double num, int digits) {...}