Chosen Languages: Python, Java, C++

1.Are the operators (e.g. /, //, **, %) used the same way?
Java: // and ** is not used in Java. Instead, Math.pow() is used for exponentiation.
Python: / is used for floating point numbers, // is used for integer division, ** is used for exponentiation, and % is used for modulus.
C++: // and ** are not used in C++. Instead, pow() is used for C++.

2.
Do string operations like + and * behave the same?

Python: uses + to link / add things together; * used to repeat or multiply
Java: uses + to link / add things together; * used to multiply, repeating must use repeat()
C++: uses + to link / add things together; * used to multiply, repeating need loop function

3.
How are variables declared and updated?

For Python, variables are declared by setting the variable name equal to a value using A single = sign (ex. x=5). For Java and C++, the variables are declared the same way But with the int() instruction in front and with a semicolon at the end. The variables are Updated in the same way across all languages (ex. x = x+5), but again, Java and C++ include A semicolon at the end.

4. Are types required, or can you assign freely?

For Java and C++, the data types are required. For python, variables can be assigned freely

5. What happens when an error occurs?

When an error occurs, all 3 programming languages output "error trace back crash".
The program stops running (terminates) in all languages.

6.
How is the error shown or handled in each language?

The python interpreter stops and displays an error message. Java creates an Exception object and C++ uses "try-catch" blocks and exceptions as well.

7.
Are true/false written the same?
For python, t = True and f = False. For java and C ++, t = true and f = false.

8.
What symbols or keywords are used for logical operations?
C++ and Java use &&, ||, and ! for the logical AND, OR, and NOT statements. Python uses and, or, not for its logical statements.

9.
How is conditional logic structured?
C++ and Java use parentheses and curly brackets for conditional logic, while Python simplifies this with colons and indentation.

10.
Are keywords or formatting different?
C++ and Java require semicolons to move on to the next statement and curly brackets for code blocks, whereas Python relies on indentation to group statements without semicolons or brackets.

11.
How does each language group lines of code (indentation, braces, etc.)?

Python groups lines of code with indentations to separate "blocks" of code and sometimes Spaces in between large amounts of code. Java and C++ indent the code as well, but also use brackets.

12.
How is a simple loop written and displayed?

A "for" loop in python starts out by assigning a variable (i) with no value, and then everytime 'i' is in some range that is defined, the program will output something. It looks like this:

```
for i in range(3):
    print('Loop', i)
```

In Java and C++, the "for" loop begins with the keyword "for", and then parentheses. Inside Those parentheses, it first assigns a value to a variable (ex. i) (int(i) = 0). Then, in the same

Parentheses, a range is set (ex. i<3) after a comma. After that, there is a line stating i++, which Means to add 1 to the value of i every time the program returns to the start, and until i does not Meet the condition stated in the range. It looks like this:

```
for (i=0, i<3, i++) {
        Std::cout << “Loop” << i << std:endl;
}
```

13.
Which language was easiest to understand?
The language that was the easiest to understand was Python.

14.
Which had the clearest output or errors?
Python provides the clearest and most detailed error messages as they are easy to read, while C++ and Java also provide error messages but can contain more jargon or potentially be harder to debug.

15.
Which would you recommend to a beginner
I would recommend learning Python for a beginner because Python provides the most detailed error messages, so it is easy to fix your mistakes.