|  |  |
| --- | --- |
| **Date Assigned: 9/19/16** | **Date Due: 9/21/16** |
| **Unit:** Methodology | **Turn In List:** **1. Terms, 2. Post timeline, and 3. Grid** |
| *“I can create and use many data types in a simple computer program.”* | |

**Data Types and Variables: A look at the major data types for modern languages**

**Content Objectives:** Students will be able to declare, initialize and assign variable for a program.

|  |
| --- |
| **Starter Activity** |
| // Consider Mr Kapptie’s grading system where numbers  // are turned into letters. Fill in the blanks in the  // following code to complete the boolean expression.  float grade = random(0,100);  if (\_\_\_\_\_\_\_) {  println("Assign letter grade A.");  } else if (\_\_\_\_\_\_\_\_) { // In one conditional statement, you can only ever have one if and one else. However, you can have as many else if's as you like!  println (\_\_\_\_\_\_\_\_);  } else if (\_\_\_\_\_\_\_\_) {  println (\_\_\_\_\_\_\_\_);  } else if (\_\_\_\_\_\_\_\_) {  println (\_\_\_\_\_\_\_\_);  } else {  println (\_\_\_\_\_\_\_\_);  }  // Create a method to use in an app to display letter grade based on the  // position of mouseX on a line. |

|  |  |
| --- | --- |
| **Key Terms:** | |
| Interpreted Language | An interpreted language is a programming language for which most of its implementations execute instructions directly, without previously compiling a program into machine-language instructions |
| Compiled Language | A compiled language is a programming language whose implementations are typically compilers, and not interpreters |
| Low Level Language | A low-level language is a programming language that provides little or no abstraction from a computer’s instruction set architecture |
| High Level Language | A high-level language is a programming language such as C, FORTAN, or Pascal that enables a programmer to write programs that are more or less independent of a particular type of computer |
| Execute | The process by which a computer or a virtual machine performs the instructions of a computer program |
| Identifiers |  |
| Declare Variables |  |
| Initialize Variables |  |
| Assign Variables |  |

|  |
| --- |
| **Assignment:** |
| For each data type give the following information. Use the Processing reference as an aid (note that all data types follow the java standard.) You may write N/A where applicable.   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | **Memory Used** | **Possible Values (Min)** | **Possible Values (Max)** | **Purpose** | **Syntax** | | boolean |  |  |  |  |  | | byte |  |  |  |  |  | | char |  |  |  |  |  | | color |  |  |  |  |  | | double |  |  |  |  |  | | float |  |  |  |  |  | | int |  |  |  |  |  | | long |  |  |  |  |  | | String |  |  |  |  |  | | XML |  |  |  |  |  | | Array |  |  |  |  |  | | ArrayList |  |  |  |  |  | | Table |  |  |  |  |  |   Create a new processing project with a medium gray canvas size of 1000 x 1000 pixels and draw a black grid on the first made up of lines at every 100 pixels vertically and horizontally. Provide text labels (100, 200, etc.) on the left margin and top margin. |

Notes (Points of interest, mistakes, lessons learned, web resources, and thoughts):

|  |
| --- |
|  |