|  |  |
| --- | --- |
|  |  |
| **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** |
| //Jacob Schwartz 9/19/18  // 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.  void setup() {  size (100, 100);  }  //float grade = random(0,100);    void draw() {  if (mouseX >= 94) {  println(mouseX + " You got an A");  } else if (mouseX >= 89) { // 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 (mouseX + " You got a A-");  } else if (mouseX >= 87) {  println (mouseX + " You got a B+");  } else if (mouseX >= 83) {  println (mouseX + " You got a B");  } else if (mouseX >= 80) {  println (mouseX + " You got a B-");  } else if (mouseX >= 77) {  println (mouseX + " You got a C+");  } else if (mouseX >= 73) {  println (mouseX + " You got a C");  } else if (mouseX >= 70) {  println (mouseX + " You got a C-");  } else if (mouseX >= 67) {  println (mouseX + " You got a D+");  } else if (mouseX >= 63) {  println (mouseX + " You got a D");  } else if (mouseX >= 60) {  println (mouseX + " You got a D-");  } else {  println (mouseX + " You got an F");  }  }  // Create a method to use in an app to display letter grade based on the  // position of grade on a line. |

|  |  |
| --- | --- |
| **Key Terms:** | |
| Interpreted Language | A language that has its source files on a computer that has an interpreter. Reads instructions form a source file without compiling or changing to machine code. ex: python, html |
| Compiled Language | A language that has its source files compiled into machine code. It is read by the OS. |
| Low Level Language | Really close to the hardware: giving instruction directly to the computer hardware. Usually in hex or binary. |
| High Level Language | Use English terms to give instruction to the compiler. |
| Execute | Moving an application to system memory. |
| Identifiers | The name given to a datatype in memory. |
| Declare Variables | Reserving the memory for that datatype. |
| Initialize Variables | Giving a variable its first value |
| Assign Variables | Changing the value in memory using the = command. |

|  |
| --- |
| **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 | 1 bit | False, 0, off | True, 1, on | Control statements for app flow | boolean a= false; | | byte | 8 bits | -128 | 127 | sending data | byte b = -128; | | char | 2 bytes (16 bits) | NA | NA | datatype for characters | char var = ‘K’; | | color | 32 bits of info | #000000 | #FFFFFF | datatype for colors | color c2  #FFCC00 | | double | 64 bit | -3.40282347E+38 | 3.40282347E+38 | datatype for floating-point numbers larger than for float | double a;  double b = -2.984D | | float | 32 bit | -3.40282347E+38 | 3.40282347E+38 | Data type for floating-point numbers (have a decimal point) | float a;  float b = -2.984; | | int | 32 bits of info | -2,147,483,648 | 2,147,483,647 | datatype for integers, numbers without decimals | int h = 0; | | long | 64 bits of info | -9,223,372,036,854,775,808 | 9,223,372,036,854,775,807 | datatype for large integers | long a;  long b = -256 | | String | NA | NA | NA | sequences of characters | String(data)  String(data, offset, length) | | XML | NA | NA | NA | holds a representation of an XML object | XML(name) | | Array | NA | NA | NA | a list of data | datatype[] var  var[element] = value  var.length | | ArrayList | NA | NA | NA | stores a variable number of objects | ArrayList<Type>()  ArrayList<Type>(initialCapacity) | | Table | NA | NA | NA | store data with multiple rows and columns, like a spreadsheet | Table()  Table(rows) |   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):

|  |
| --- |
|  |