|  |  |
| --- | --- |
| **Date Assigned: 1/5/16** | **Date Due: 1/7/16** |
| **Unit:** Languages | **Turn In List:** **1. Terms** |
| *“I will be able to declare the language of focus for Semester 2 .”* | |

**Computer Programming Languages: An in-depth analysis**

**Content Objectives:** Students will research each of the five languages acceptable for the 2A computer programming state CTE certification. The following [Wiki article](http://en.wikipedia.org/wiki/Comparison_of_programming_languages) may help in your search. [Language popularity article](http://en.wikipedia.org/wiki/Measuring_programming_language_popularity).

|  |
| --- |
| **Starter Activity** |
| Write a class that will run in Processing. You may choose from the following list of class names: Human, Cat, Dog, Spaceship, Soldier or Planet. The class must contain a name, at least 2 class variables, 1 constructor, a display function and at least one action function. Paste code below:  class Planet {  float x, y, w, h;  color c;  Button(float tempX, float tempY, float tempW, float tempH, color tempC) {  x = tempX;  y = tempY;  w = tempW;  h = tempH;  c = tempC;  }  void printColor()  {  println(c);  }  } |

|  |  |
| --- | --- |
| **Key Terms: (lookup each language and write a short description of each)** | |
| **C++** | has [imperative](https://en.wikipedia.org/wiki/Imperative_programming), [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) and [generic](https://en.wikipedia.org/wiki/Generic_programming) programming features, while also providing facilities for low-level memory manipulation. |
| **C#** | [multi-paradigm programming language](https://en.wikipedia.org/wiki/Multi-paradigm_programming_language) encompassing [strong typing](https://en.wikipedia.org/wiki/Strong_typing), [imperative](https://en.wikipedia.org/wiki/Imperative_programming), [declarative](https://en.wikipedia.org/wiki/Declarative_programming),[functional](https://en.wikipedia.org/wiki/Functional_programming), [generic](https://en.wikipedia.org/wiki/Generic_programming), [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) ([class](https://en.wikipedia.org/wiki/Class_(computer_science))-based), and [component-oriented](https://en.wikipedia.org/wiki/Component-based_software_engineering) programming disciplines. It was developed by [Microsoft](https://en.wikipedia.org/wiki/Microsoft) within its [.NET](https://en.wikipedia.org/wiki/.NET_Framework) initiative |
| **Java** | a general-purpose [computer programming language](https://en.wikipedia.org/wiki/Programming_language) that is [concurrent](https://en.wikipedia.org/wiki/Concurrent_computing), [class-based](https://en.wikipedia.org/wiki/Class-based_programming), [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming),[[12]](https://en.wikipedia.org/wiki/Java_(programming_language)#cite_note-FOOTNOTEGoslingJoySteeleBracha20141-12) and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "[write once, run anywhere](https://en.wikipedia.org/wiki/Write_once,_run_anywhere)" |
| **Python** | general, high-level language designed for readability and usable with many paradigms. |
| **Visual Basic** | **Java** is a general-purpose [computer programming language](https://en.wikipedia.org/wiki/Programming_language) that is [concurrent](https://en.wikipedia.org/wiki/Concurrent_computing), [class-based](https://en.wikipedia.org/wiki/Class-based_programming), [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming),[[12]](https://en.wikipedia.org/wiki/Java_(programming_language)#cite_note-FOOTNOTEGoslingJoySteeleBracha20141-12) and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "[write once, run anywhere](https://en.wikipedia.org/wiki/Write_once,_run_anywhere)" |
| Type Safety | the extent to which a [programming language](https://en.wikipedia.org/wiki/Programming_language) discourages or prevents **type errors**. A type error is erroneous or undesirable program behaviors caused by a discrepancy between differing [data types](https://en.wikipedia.org/wiki/Data_type) for the program's constants, variables, and methods |
| Interpreted | a programming **language** for which most of its implementations execute instructions directly, without previously compiling a program into machine-**language** instructions. |
| Procedural | relies on procedures, also known as routines or subroutines. A **procedure** contains a series of computational steps to be carried out |
| Compiled | are compiled into bytecode, the virtual machine-friendly interpreted language. **Lisp** implementations can freely mix interpreted and compiled code. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **C++** | **C#** | **Java** | **Python** | **Visual Basic** |
| Intended Use | Fast, more machine-level commands | For windows applications, visual commands | Write once, run on all devices. run on virtual machine | Have less and more readable code, multiple paradigms | easy to learn, visuals |
| Strongly Typed | yes | yes | yes | no | no |
| OS’s | mac, windows, ect. | windows | all | windows, mac, linux | windows, mac |
| Industry | intensive, low level tasks- video | Windows applications | Many things | web development | old projects and maintenance of them |
| **Atoms or Bits** | atoms | bits | bits | bits | bits |
| Current Version | 14 | 6 | 8 | 8 | 6 |
| Official Standard | https://isocpp.org/std/the-standard | https://msdn.microsoft.com/en-us/library/ms228593.aspx | http://docs.oracle.com/javase/specs/ | https://www.python.org/dev/peps/pep-0008/ | https://msdn.microsoft.com/en-us/library/h63fsef3.aspx |

|  |
| --- |
| **History and Background of the Language you are interested in:** |
| You may work in pairs for this portion but you need to submit your own file to Canvas. Give the When’s, Who’s, Why’s, Where’s, How’s and worldwide popularity pulse applicable for the language you are considering. (Note, this is NOT your final decision.)  **Python** |

|  |
| --- |
| **Assignment:** |
| Rewrite Class from Starter:  Find the official standard website or simply do a google search for your language and “class” or “object” and do your best to re-write the class from starter in the new language (code not require to build or compile.)  class Spaceship:   def \_\_init\_\_(self, x, y, w, h, c):  self.x = x  self.y = y  self.w = w  self.h = h  self.c = c  def drawIt(self):  print self.c |

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

|  |
| --- |
|  |