**Comparisons between C++ vs JAVA vs Swift languages**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **C++** | **JAVA** | **SWIFT** |
| **Primitive Data Type** | - bool, char(1 byte), unsigned int, int(4), short int(2), long int(4), long long int(8byte), float(4), double(8), long double, w\_char. | -boolean, byte, char(16 bits), short(16 bits), int, long(64 bits), float, double. | -Bool, Character, UInt, Int8, Int16, Int32, Int64, Float, Double  -String  -Optional  -Tuples |
| **Non-Primitive:** | Pointer: \*,\*\*, [3][4]  vector, set, map | Pointer [][]  -java.lang.String, Arrays, List, Vector, LinkedList, HashMaps, HashTable, Stack, Queue  -Set | -Array([[Int]), Dictionary([String: String]), Set |
| **For-Loop** | for (int i = 0; i < n; i++)  for (int i : array) | for (int i = 0; i < n; i++)  for (int i : array) | for var I in 0..<n  for (let i in array) |
| **Switch-case** | Need break to avoid going to the next case | Need break to avoid going to the next case | Noneed break to avoid going to the next case  User fallthrough to go to the next case |
| **Goto** | Has goto | break Label: go to label | No goto |
| **Generic** | template | <T> | <T: Base\_Class/Protocol> |
| **Enum** | -No function | Has functions  Has value (Red(“Stop”)  .name() | Has function(mutating)  Has value(red = “Stop”)  rawValue |
| **Struct** | Most similar to class except struct has default public access modifiers  Struct has inheritance. | No Struct | Struct is value type while class is reference type(mutating)  Struct can not be inherited |
| **Abstract class** | Cannot be an object  Has virtual function | Cannot be an object  Has abstract function | Protocol: Cannot be an object  Extension for concrete function |
| **Interface** | No | Not be an object, no constructor  All methods are abstract & public.  All attributes are public, static, final | No |
| **Constant** | const | final(initialization in constructor) | let |
| **Access modifiers** | Private, protected, public  Classes have not | Default(for the same package, not for other package subclass), Private, protected(for the same package, for other package subclass), public(for the same & other package)  Classes have | open(subclass outside module), public(subclass within module), internal(default, for inside module), fileprivate, private  Class have |
|  |  |  |  |
| **Inheritance**  **-Constructor**  **-Access modifiers**  **-Parent pointer**  **-Self pointer** | Multiple  No required+auto calling  Changeable in overriden methods  super  this | Single  Required+auto calling  Non- Changeable in overriden methods  super  this | Single  No required+auto calling  Non- Changeable in overriden methods  super  self |
| **Multiple thread**  **Sharing: Memory or resource** | std::thread for function, object & lambda pointers  std::mutext + std:lock\_guard | Thread: extend  Runable: implement  -synchronized keyword for locking class, method or block of code.  -volatile keyword for variable | NSOperationQueue  GCD  -Using gcd with barrier flag |

**Private properties of each language**

**C++:**

-Friend: function, class

-Pointer: new, delete, delete[]

-Stack memory: organize by CPU. It is fast but its size is limited.

-Heap memory: slower than Stack memory, but no limitation at size theorically.

**Java:**

-No operator overriding

-Static block: run one time when initializing class

-Static class: unable to be derived

-Static methods: unable to be overriden

**Swift:**

-Closure

-Optional: wildcast(\_), chaining(a?.b?.c),

-Type casting: if let

**References:**

**C++:**

-Multi-Threading:

<https://hackernoon.com/learn-c-multi-threading-in-5-minutes-8b881c92941f>

<https://www.quora.com/What-makes-C++-really-difficult-to-learn>

**Java:**

-Concurrency(multi-threading):

<https://www.vogella.com/tutorials/JavaConcurrency/article.html>