

<b>Keywords(Java)</b>	<b>Kryptonic Keywords</b>
<u><b>Data Type:</b></u> int float string bool	Int float string bool
<u><b>Conditional Statement:</b></u> If/else/else if	if/else/elif
<u><b>Iterative Loops:</b></u> for while switch	for while switch
<u><b>Object Oriented Programming:</b></u> class extends this super static new interface final	class child_of present super static new interface fixed
<u><b>Access Modifiers:</b></u> public private	shared own

## **Punctuators:**

[ ] { } ( ) , . ;

## **Operators:**

+ - / \* = > < >= <= == += -= \*= != !% ++ --

<b><u>Variable Declaration</u></b>	take
<b><u>Function Declaration</u></b>	\$function
<b><u>Main Function Declaration</u></b>	\$mfunction
<b><u>Return Type</u></b>	Declared after naming function return (inside the function)

## **Rules & Syntax:**

### **Comments:**

# Single Line Comment #\*

Multi Line Comment #??#

### **Variable Declaration:**

take String :

name;

OR

take String: name = "abc";

### **Function Declaration:**

\$function numberAddition int (int a, int b)

{ take int: c = a + b;

return c;

}

\$function printNumberAddition void (int a, int b)

{ take int: c = a + b;

print c;

}

\$mfunction main void () {

#body

}

## **Conditional Statement:**

```
if (condition)
```

```
{ #body
```

```
} else
```

```
{ #body
```

```
} elseif
```

```
(condition)
```

```
{ #body
```

```
}
```

## **Iterators / Loops:**

```
for(take int: i = 0; i > 10; i++){
```

```
#body
```

```
}
```

```
while(condition){
```

```
#body
```

```
}
```

## Object Oriented Programming:

```
class Fruit {
```

```
  take string: name;
```

```
}
```

```
class Apple implements A child_of Fruit {
```

```
  take string: name;
```

```
  present.name = "apple"
```

```
}
```

```
interface class A {
```

```
  take string: name;
```

```
  $function getName void();
```

```
}
```