

```

fun <T> everyNth(L: List<T>, N: Int): List<T> {
    /* N is given to be a positive integer */
    val Nth: Int = N - 1
    val MAX: Int = L.size - 1
    /* L can be any type so we use a supertype */
    var Nth_list: MutableList<T> = mutableListOf<T>()
    /* for(i = N-1; i < L.size(); i += N) */
    for(index in Nth..MAX step N) {
        Nth_list.add(L.get(index))
    }

    /* Convert our MutableList to a regular List
     * Kotlin's regular lists are readonly, hence immutable */
    val final_list: List<T> = Nth_list.toList()
    return final_list
}

fun main() {
    println("Starting testing...")

    println()
    /* List of Ints */
    val L1: List<Int> = listOf(1,2,3,4,5,6,7,8,9,10)
    val N1: Int = 2
    val A1: List<Int> = listOf(2,4,6,8,10)
    val B1: List<Int> = everyNth(L1,N1)
    if(B1.equals(A1)) {
        println("Test1 successful.")
        println("L = " + L1)
        println("N = " + N1)
        println("Returned " + B1)
    }
    else {
        println("Test1 failed.")
        println("The list L = " + L1 + " with N = " + N1 + " yielded:" )
        println(B1)
    }

    println()
    /* List of Strings */
    val L2: List<String> =
listOf("a","b","c","d","e","f","g","h","i","j")
    val N2: Int = 2
    val A2: List<String> = listOf("b","d","f","h","j")
    val B2: List<String> = everyNth(L2,N2)
    if(B2.equals(A2)) {
        println("Test2 successful.")
        println("L = " + L2)
        println("N = " + N2)
        println("Returned " + B2)
    }
    else {
        println("Test2 failed.")
        println("The list L = " + L2 + " with N = " + N2 + " yielded:" )
        println(B2)
    }

    println()
    /* List of Doubles */

```

```
    val L3: List<Double> =  
listOf(1.1,2.2,3.3,4.4,5.5,6.6,7.7,8.8,9.9,10.0)  
    val N3: Int = 2  
    val A3: List<Double> = listOf(2.2,4.4,6.6,8.8,10.0)  
    val B3: List<Double> = everyNth(L3,N3)  
    if(B3.equals(A3)) {  
        println("Test3 successful.")  
        println("L = " + L3)  
        println("N = " + N3)  
        println("Returned " + B3)  
    }  
    else {  
        println("Test3 failed.")  
        println("The list L = " + L3 + " with N = " + N3 + " yielded:" )  
        println(B3)  
    }  
  
    println()  
    println("Testing concluded.")  
}
```