

# Demonstrate a generic interface



```
using System;

public interface ISequence {
    T getNext();
    void reset();
    void setStart(T v);
}

class ByTwos : ISequence {
    T start;
    T val;

    public delegate T IncByTwo(T v);

    IncByTwo incr;

    public ByTwos(IncByTwo incrMeth) {
        start = default(T);
        val = default(T);
        incr = incrMeth;
    }

    public T getNext() {
        val = incr(val);
        return val;
    }

    public void reset() {
        val = start;
    }

    public void setStart(T v) {
        start = v;
    }
}
```

```
val = start;
}
}

class GenIntfDemo {
static int intPlusTwo(int v) {
return v + 2;
}

static double doublePlusTwo(double v) {
return v + 2.0;
}

public static void Main() {
ByTwos intBT = new ByTwos(intPlusTwo);

for(int i=0; i < 5; i++) Console.Write(intBT.getNext() + " ");
Console.WriteLine(); ByTwos dblBT = new ByTwos(doublePlusTwo);

dblBT.setStart(11.4);

for(int i=0; i < 5; i++) Console.Write(dblBT.getNext() + " ");
} } [/csharp]
```