1. Demonstration a generic class with one parameter.

```
class Pair<T>
{
   public Pair() { first = null; second = null; }
   public Pair(T first, T second) { this.first = first;
this.second = second; }

   public T getFirst() { return first; }
   public T getSecond() { return second; }

   public void setFirst(T newValue) { first = newValue; }
   public void setSecond(T newValue) { second = newValue; }

   private T first;
   private T second;
}

class ArrayAlg
{
    /**
       Gets the minimum and maximum of an array of strings.
       @param a an array of strings
       @return a pair with the min and max value, or null if a is
       null or empty
    */
    public static Pair<String> minmax(String[] a)
    {
       if (a == null || a.length == 0) return null;
       String min = a[0];
       String max = a[0];
       for (int i = 1; i < a.length; i++)
       {
          if (min.compareTo(a[i]) > 0) min = a[i];
          if (max.compareTo(a[i]) < 0) max = a[i];
       }
       return new Pair<String>(min, max);
    }
 }

public class PairTest1
  {
     public static void main(String[] args)
     {
        String[] words = { "Mary", "had", "a", "little", "lamb"
};
        Pair<String> mm = ArrayAlg.minmax(words);
        System.out.println("min = " + mm.getFirst());
        System.out.println("max = " + mm.getSecond());
     }
 }
```

The result:

2. Demonstrate a simple generic method.

```
/*
Java 2, v5.0 (Tiger) New Features
by Herbert Schildt
ISBN: 0072258543
Publisher: McGraw-Hill/Osborne, 2004
*/

public class GenMethDemo {

  // Determine if an object is in an array.
  static <T, V extends T> boolean isIn(T x, V[] y) {

    for(int i=0; i < y.length; i++)
      if(x.equals(y[i])) return true;

    return false;
  }

  public static void main(String args[]) {

    // Use isIn() on Integers.
    Integer nums[] = { 1, 2, 3, 4, 5 };

    if(isIn(2, nums))
      System.out.println("2 is in nums");

    if(!isIn(7, nums))
      System.out.println("7 is not in nums");

    System.out.println();

    // Use isIn() on Strings.
    String strs[] = { "one", "two", "three",
                      "four", "five" };

    if(isIn("two", strs))
      System.out.println("two is in strs");

    if(!isIn("seven", strs))
      System.out.println("seven is not in strs");

    // Opps! Won't compile! Types must be compatible.
//    if(isIn("two", nums))
//      System.out.println("two is in strs");
  }
}
```

The result:

```
D:\Java_Dev\WEB\java2s>java GenMethDemo
2 is in nums
7 is not in nums
```

3. Split a String into a Java Array of Strings divided by an Regular Expressions

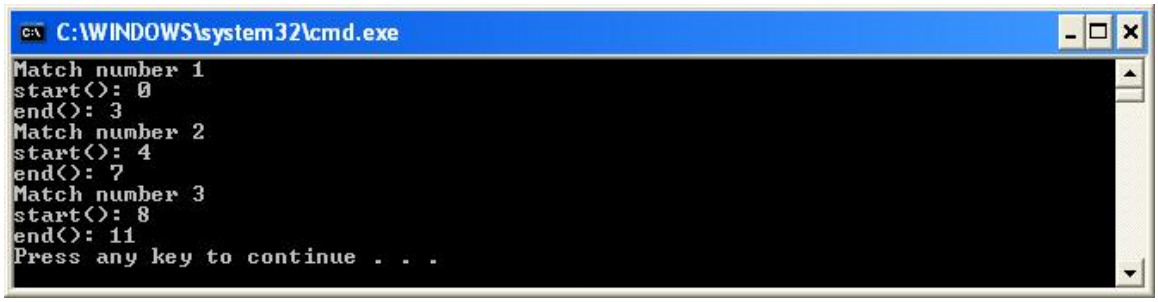
```
Import java.util.regex.*;

/** Split a String into a Java Array of Strings divided by an RE
 */
public class Split {
  public static void main(String[] args) {
    String[] x =
      Pattern.compile("ian").split(
        "the darwinian devonian explodian chicken");
    for (int i=0; i<x.length; i++) {
      System.out.println(i + " \"" + x[i] + "\"");
    }
  }
}
```

The result:

4. Find a string using pattern

```
import java.util.regex.Matcher;
import java.util.regex.Pattern;

public final class MatcherTest {

  private static final String REGEX = "\\bdog\\b";

  private static final String INPUT = "dog dog dog doggie dogg";

  public static void main(String[] argv) {
    Pattern p = Pattern.compile(REGEX);
    Matcher m = p.matcher(INPUT); // get a matcher object
    int count = 0;
    while (m.find()) {
      count++;
      System.out.println("Match number " + count);
      System.out.println("start(): " + m.start());
      System.out.println("end(): " + m.end());
    }
  }
}
```

The result:



Do It Yourself

5.1. Do workshop of the module 6, 7

5.2. Create a class Car with fields: Name, Price, Production and properly methods. Create another class named GenericCar with a parameter of the T type. This class manages a collection of object T (may be LinkedList). Implementing some methods for GenericCar: add, display, getSize, checkEmpty, delete.
Write a program to use GenericCar as below menu (use Regular expression to validate data: Name doesn't contain any numeric character, Price contains only numeric characters).

```
Menu
------------------------
Add
Display
GetSize
CheckEmpty
Grad
Exit
Your choice: _
```

5.3. Read content of a text file replace all the word ‘is’ into ‘was’ and display after-effected text on screen.

References

+ Java tutorials

+ Javadoc

+ Java2s.com

+ Javapassion.com

+ Java almanac
http://www.exampledepot.com