Passing Arrays to Methods
(resembles "pass-by-reference")

Arguments/parameters and methods are passed -*by-value*.
Copies of argument values are sent to the method, where the copy is manipulated and in certain cases, one value may be returned.
While the copied values may change in the method, ***the original values*** in main do **NOT** change (unless purposely reassigned **after** the method).

Passing an array to methods is *"pass-by-reference",* meaning that when an array is passed as an argument, its **memory address location** (its "reference") is used.
In this way, the contents of an array CAN be changed inside of a method, since we are dealing directly with the actual array and not with a copy of the array.

```
public class FindSum
{
        public static void main (String [ ] args)
        {
  int [] number = {1,2,3,4,5,6,7,8,9,10};
  int i;
  int sum=0;

  sum = find_sum(number);   // invoke the method
  System.out.println("The sum is  " +sum + ".");
        }
public static int  find_sum(int [] value)  //method
definition to find sum
      {
            int i, total = 0;
            for(i=0; i<10; i++)
            {
                total = total + value[i];
            }

            return (total);
        }


}
```