Array

- An array is a collection of variables that are accessed with an index number.
- Arrays in the C++ programming language Arduino sketches are written in and can be complicated, but using simple arrays is relatively straightforward.
- Arrays are zero indexed, that is, referring to the array initialization above, the first element of the array is at index 0.

`mySensVals[0] == 2, mySensVals[1] == 4`, and so forth.

It also means that in an array with ten elements, index nine is the last element. Hence:

```
int myArray[10]={9, 3, 2, 4, 3, 2, 7, 8, 9, 11};
// myArray[9]    contains 11
// myArray[10]   is invalid and contains random information (other
memory address)
```

while

- while loops will loop continuously, and infinitely, until expression inside the parentheses becomes false.
- Something must change the tested variable, or the while loop will never exit.
- This could be in your code, such as an incremented variable, or an external condition, such as testing a sensor.

```
while  (someVariable  ?? value)

{

     doSomething;

}
```

The following example test whether ‘someVariable’ is less than 200 and if true executes the statements inside the brackets and will continue looping until ‘someVariale’ is no longer less than 200.

```
While (someVariable < 200)  // test if less than 200

{

     doSomething;          // executes enclosed statements
```

```
      someVariable++ ;     // increments variable by 1
}
```

do ... while

- The do loop is a button driven loop that works in the same manners as the while loop, with the exception that the condition is tested at the end of the loop, so the do loop will always run at least once.

```
do
{
     doSomething;
}    while (someVariable  ?? value);
```

The following examples assigns readSensors() to the variable ‘x’, pauses for 50 milliseconds, then loops indefinitely until ‘x’ is no longer less than 100:

```
do
{
     x = readSensors();    // assigns the value of readSensors() to x
     delay (50);           // pauses 50 milliseconds
}    while (x < 100); // loops if x is less than 100
```