

**Version 0: Collect 4 grades for a semester and add them together without arrays**

Without using arrays, we need a **variable for each mark**. Whenever we need to something with the marks, we need to refer to each of them by name, one at a time.

```
// Student grades for one semester (4 grades)
class ArrayDemo0
{
    public static void main(String[] args)
    {
        // without arrays
        double mark1, mark2, mark3, mark4;
        double sum;    // total of all marks

        // get first grade
        System.out.println("Mark 1?");
        mark1 = In.getDouble();

        // get the remaining grades
        System.out.println("Mark 2?");
        mark2 = In.getDouble();
        System.out.println("Mark 3?");
        mark3 = In.getDouble();
        System.out.println("Mark 4?");
        mark4 = In.getDouble();

        // output marks
        System.out.println("Mark 1 = " + mark1);
        System.out.println("Mark 2 = " + mark2);
        System.out.println("Mark 3 = " + mark3);
        System.out.println("Mark 4 = " + mark4);

        // add marks together
        sum = mark1 + mark2 + mark3 + mark4;

        // output sum
        System.out.println("Sum = " + sum);
    }
}
```

**Version 1: Collect 4 grades for a semester and add them together using an array**

By using an array, we only need a **single variable** (although this variable **needs to have 4 spots** to hold data). Not much else changes (yet), since we still need to **refer to each part of the array** to access the data.

```
// Student grades for one semester (4 grades)
class ArrayDemol
{
    public static void main(String[] args)
    {
        // with arrays
        double mark[];
        mark = new double[4];
        double sum;    // total of all marks

        // get first grade
        System.out.println("Mark 1:");
        mark[0] = In.getDouble();

        System.out.println("Mark 2:");
        mark[1] = In.getDouble();
        System.out.println("Mark 3:");
        mark[2] = In.getDouble();
        System.out.println("Mark 4:");
        mark[3] = In.getDouble();

        // output marks
        System.out.println("Mark 1 = " + mark[0]);
        System.out.println("Mark 2 = " + mark[1]);
        System.out.println("Mark 3 = " + mark[2]);
        System.out.println("Mark 4 = " + mark[3]);

        // add marks together
        sum = mark[0] + mark[1] + mark[2] + mark[3];

        // output sum
        System.out.println("Sum = " + sum);
    }
}
```

**Version 2: Using an array for 4 grades and loops for control**

Each time we refer to an element of the array, we cycle through the values 0, 1, 2, 3 (which is a total of 4 elements). This repetition of 0 to 3 should bring to mind the idea of a loop – in particular, the FOR loop (which is for counting).

**Each time the loop executes, it counts from 0 to 3.** The value of the count is contained in the variable *i*, which we use as the **index of the array**.

```
// Student grades for one semester (4 grades)
class ArrayDemo2
{
    public static void main(String[] args)
    {
        // with arrays and loops
        double mark[];
        mark = new double[4];
        double sum;    // total of all marks

        // get grades
        for (int i = 0; i <= 3; i++)
        {
            System.out.println("Mark " + (i+1) + "?");
            mark[i] = In.getDouble();
        }

        // output marks
        for (int i = 0; i <= 3; i++)
        {
            System.out.println("Mark " + (i+1) + " = " + mark[i]);
        }

        // add marks together
        sum = 0;
        for (int i = 0; i <= 3; i++)
        {
            sum = sum + mark[i];
        }

        // output sum
        System.out.println("Sum = " + sum);
    }
}
```

**Version 3: Using an array for 10 grades and loops for control**

Another advantage of using arrays is that we can easily change the size of the overall array quite quickly. Here, we **change the number of grades from 4 to 10** with only a few modifications to the code.

```
// Student grades for one semester (10 grades)
class ArrayDemo3
{
    public static void main(String[] args)
    {
        // with arrays and loops
        double mark[];
        mark = new double[10];
        double sum;    // total of all marks

        // get grades
        for (int i = 0; i <= 9; i++)
        {
            System.out.println("Mark " + (i+1) + "?");
            mark[i] = In.getDouble();
        }

        // output marks
        for (int i = 0; i <= 9; i++)
        {
            System.out.println("Mark " + (i+1) + " = " + mark[i]);
        }

        // add marks together
        sum = 0;
        for (int i = 0; i <= 9; i++)
        {
            sum = sum + mark[i];
        }

        // output sum
        System.out.println("Sum = " + sum);
    }
}
```

**Version 4: Using an array for 15 grades and loops for control, only changing a single value**

If the code is written carefully, it is possible to make a **single variable** control **all references to the size of the array**. If the size has to be changed in the future, a single modification is enough.

```
// Student grades for one semester
class ArrayDemo4
{
    public static void main(String[] args)
    {
        // with arrays and loops
        double mark[];
        int length = 15;
        mark = new double[length];
        double sum;    // total of all marks

        // get grades
        for (int i = 0; i <= length-1; i++)
        {
            System.out.println("Mark " + (i+1) + "?");
            mark[i] = In.getDouble();
        }

        // output marks
        for (int i = 0; i <= length-1; i++)
        {
            System.out.println("Mark " + (i+1) + " = " + mark[i]);
        }

        // add marks together
        sum = 0;
        for (int i = 0; i <= length-1; i++)
        {
            sum = sum + mark[i];
        }

        // output sum
        System.out.println("Sum = " + sum);
    }
}
```

**Version 5: Let the user decide how long the array should be**

It is possible to **ask the user how big the array should be**, and then **create the array**.

```
// Student grades - ask user how many grades
class ArrayDemo5
{
    public static void main(String[] args)
    {
        // with arrays and loops
        double mark[];
        int length;    // array length from user
        double sum;    // total of all marks

        // this is one of the few good reasons to declare variables
        // later in the code
        System.out.println("How many grades will you enter?");
        length = In.getInt();

        // create an array according to user input
        mark = new double[length];

        // get grades
        for (int i = 0; i <= length-1; i++)
        {
            System.out.println("Mark " + (i+1) + "?");
            mark[i] = In.getDouble();
        }

        // output marks
        for (int i = 0; i <= length-1; i++)
        {
            System.out.println("Mark " + (i+1) + " = " + mark[i]);
        }

        // add marks together
        sum = 0;
        for (int i = 0; i <= length-1; i++)
        {
            sum = sum + mark[i];
        }

        // output sum
        System.out.println("Sum = " + sum);
    }
}
```

**Version 6: Array is larger than needed, user terminates entry with special value**

Sometimes it isn't practical for the user to specify the length of the array. For those cases, the easiest solution is to **make our array is longer than we need**. When the user enters data, they can **use a special value to indicate they are done** (leaving part of the array empty).

For this to work, we need to keep track of how many elements are in the array. **As the user enters data, we count each entry.**

Every array has a length, and you can **ask Java for this length by adding ".length" to the end of the array name**.

```
class ArrayDemo6
{
    public static void main(String[] args)
    {
        // with arrays and loops
        double mark[];
        mark = new double[100];

        double sum = 0.0;    // total of all marks
        int count = 0;      // track number of grades in array so far
        double newGrade;

        System.out.println("Enter grades for summation.");
        System.out.println("Enter a negative value to stop.");
        System.out.println("Limit of " + mark.length + " grades.");

        // get grades
        do
        {
            System.out.println("Mark " + (count+1) + "?");
            newGrade = In.getDouble();
            if (newGrade >=0)
            {
                mark[count] = newGrade;
                count = count + 1;
            }
        } while (newGrade >=0 && count < mark.length);

        // output marks, don't output empty spaces in array
        // calculate sum at the same time
        for (int i = 0; i < count; i++)
        {
            System.out.println("Mark " + (i+1) + " = " + mark[i]);
            sum = sum + mark[i];
        }

        // output sum
        System.out.println("Total of " + count + " grades entered.");
        System.out.println("Sum = " + sum);
    }
}
```

**Version 7: Dynamic array that grows as the user enters more data**

The most complicated option, and also very inefficient. Each time the user enters new data, the array is increased in size by 1. **A new array is created that is one element longer than the old array. The contents of the old array are copied to the new array** (leaving one blank spot at the end), and the **new data is put in that last spot.**

To improve efficiency, it is possible to add more than one extra space (e.g., 10 at a time), and only add them each time we run out of space.

```
class ArrayDemo7
{
    public static void main(String[] args)
    {
        // with arrays and loops
        double mark[];
        int newLength = 0; // starting length of array
        mark = new double[newLength]; // start with a length of zero
        double tempMark[]; // temporary array for dynamic sizing
        double sum = 0; // total of all marks
        int count = 0; // track number of grades in array so far
        double newGrade;

        System.out.println("Enter grades for summation.");
        System.out.println("Enter a negative value to stop.");
        System.out.println("There is no limit on the number of grades.");

        // get grades
        do
        {
            System.out.println("Mark " + (count+1) + "?");
            newGrade = In.getDouble();
            if (newGrade >=0)
            {
                // create an extra space in the array
                newLength = mark.length + 1;

                // create a temporary array with the new length
                tempMark = new double[newLength];

                // copy the old array to the temp array
                for (int i = 0; i < mark.length; i++)
                {
                    tempMark[i] = mark[i];
                }

                // put the latest data item in the temp array
                tempMark[count] = newGrade;
                count = count + 1;

                // reassign the name of the temp array
                mark = tempMark;
            }
        } while (newGrade >=0);
    }
}
```

```
// output marks, don't output empty spaces in array
for (int i = 0; i < mark.length; i++)
{
    System.out.println("Mark " + (i+1) + " = " + mark[i]);
}

// add marks together
sum = 0;
for (int i = 0; i < mark.length; i++)
{
    sum = sum + mark[i];
}

// output sum
System.out.println("Total of " + count + " grades entered.");
System.out.println("Sum = " + sum);
}
}
```