Lesson 23 Multidimensional Array

In the last two lessons, we learned about **1D** and **2D arrays**.

If we want to store data for 100 students, we'll use a **1D array**.

Instead, if we have 4 groups of 25 students each, we'll use a **2D array**.

In this lesson, we'll discuss **multidimensional arrays.**

If there are 5 schools with 10 classes each, and each class has 20 students, we want to store data for each one of them. Here, we'll use a **3D array.**

- - > We can use 4D, 5D, 6D, etc. arrays when needed.

When we use a **1D array**, then **only one** for loop is used.

When we use a **2D array**, then **two** for loops are used.

When we use a **3D array**, then **three** for loops are used.

When we use a **6D array**, then **six** for loops are used.

Example:

A program for 5 schools with 10 classes each, and each class has 20 students.

```
#include <stdio.h>
int main() {
	int i, j, k, x[5][10][20];
	for (i = 0; i < 5; i++)
		for (j = 0; j < 10; j++)
			for (k = 0; k < 20; k++)
				scanf("%d ", &x[i][j][k]);

	for (i = 0; i < 5; i++) {
		for (j = 0; j < 10; j++) {
			for (k = 0; k < 20; k++) {
				printf("%d\t", x[i][j][k]);
			}
			printf("\n");
		}
		printf("\n");
	}
}
```

Try to code yourself:

- - > click here: [Lesson 23 Multidimensional Array - Replit]

Here, three nested for loops are used:

- The **outer loop** accesses the **first** dimension of the array, which is the number of schools ⟶ 5 iterations

- The **middle loop** accesses the **second** dimension of the array, which is the number of classes ⟶ 10 iterations

- The **innermost accesses** the **third** dimension of the array, which is the number of students ⟶ 20 iterations