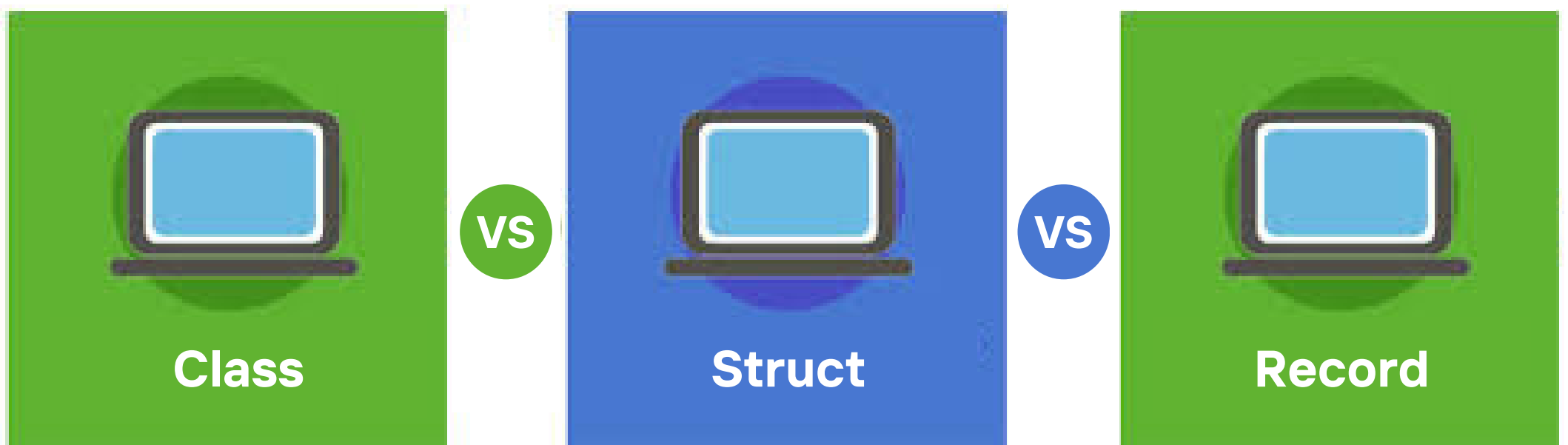


Difference Between Class, Struct, Record



Asim Noor
 [iasimnoor](https://www.linkedin.com/in/iasimnoor)

Swip



Class (Reference Type)

Overview

- Instances are always allocated on the heap.
- Supports inheritance, polymorphism, and hierarchies.
- Passed by reference, enabling shared modifications.
- Garbage collected, which may impact performance.



Asim Noor
 [iasimnoor](https://www.linkedin.com/in/iasimnoor)

Swip



Class (Reference Type)

Snippet



```
class Person
{
    public string Name { get; set; }
}
```

**Main**

```
Person p1 = new Person { Name = "Alice" };

Person p2 = p1; // Both p1 and p2 point to the same object in Heap.
p2.Name = "Bob";

Console.WriteLine(p1.Name); // Output: Bob (same object modified)
```



Asim Noor
 [iasimnoor](https://www.linkedin.com/in/iasimnoor)

Swip



Struct (Value Type)

Overview

- Stored in stack for fast allocation and access.
- No inheritance, making it lightweight and efficient.
- Passed by value, preventing unintended changes.
- Ideal for small, performance-critical data types.



Asim Noor
 [iasimnoor](https://www.linkedin.com/in/iasimnoor)

Swip



Struct (Value Type)

Snippet



```
struct Point
{
    public int X { get; set; }
    public int Y { get; set; }
}
```

**Main**

```
Point p1 = new Point { X = 10, Y = 20 };

Point p2 = p1; // A copy is created
p2.X = 50;

Console.WriteLine(p1.X); // Output: 10 (original remains unchanged)
```



Asim Noor
 [iasimnoor](https://www.linkedin.com/in/iasimnoor)

Swip



Record (Introduced in C# 9)

Overview

- Immutable by default with value-based equality.
- Supports with expressions for non-destructive updates.
- Used for DTOs to ensure consistency and reliability.
- Available as both class and struct for flexibility.



Asim Noor
 [iasimnoor](https://www.linkedin.com/in/iasimnoor)

Swip



Record (Introduced in C# 9)

Snippet

Class

```
record Person(string Name);
```

Struct

```
record struct Point(int X, int Y);
```

Main

```
Person p1 = new("Alice");  
  
Person p2 = p1 with { Name = "Bob" }; // Creates a new instance  
  
Console.WriteLine(p1.Name); // Output: Alice (original remains unchanged)
```

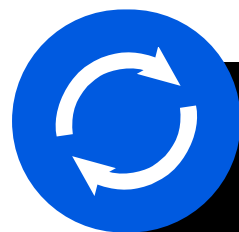


Asim Noor
 [iasimnoor](https://www.linkedin.com/in/iasimnoor)

Swip



Follow me to
get more
Information
and tips like
this.



REPOST



Asim Noor

 [iasimnoor](https://www.linkedin.com/in/iasimnoor)

