

**Override** کردن متدها یک مفهوم اساسی در برنامه‌نویسی شیء‌گرا است که به برنامه‌نویسان این امکان را می‌دهد که عملکرد متدی را که در کلاس والدین (پدری) تعریف شده است، در کلاس‌های فرزند (زیرکلاس‌ها) ، بازنویسی (**Override**) یا تغییر دهند. در این مقاله، به توضیح مفهوم **Override**، نیاز به استفاده از آن، و مثال‌هایی از کاربرد آن در زبان‌های برنامه‌نویسی مختلف می‌پردازیم.

مفهوم **Override**، به معنی جایگزینی یک متد موجود با یک متد **Override** جدید در کلاس فرزند است. وقتی یک متد در کلاس فرزند **Override** می‌شود، هرگونه فراخوانی به آن متد در کلاس فرزند به جای متد اصلی کلاس والدین انجام می‌شود.

### نیاز به استفاده از **Override**

استفاده از **Override** می‌تواند در موارد زیر مفید باشد:

1. **تعمیم عملکرد:** وقتی که نیاز است که عملکرد یک متد در کلاس فرزند متفاوت باشد.
2. **پیاده‌سازی متدهای **Abstract**:** وقتی که یک کلاس از یک کلاس **Abstract** ارث بری می‌کند، باید تمام متدهای **Abstract** آن را **Override** کند.
3. **بهبود عملکرد:** وقتی که نیاز است که عملکرد یک متد را بهبود دهید یا به نیازهای خاص کلاس فرزند تطبیق دهید

## مثال‌های Override در زبان‌های برنامه‌نویسی مختلف:

```
public class Animal
```

```
{
```

```
public virtual void MakeSound()
```

```
{
```

```
Console.WriteLine("Animal is making a sound.");
```

```
}
```

```
}
```

```
public class Dog : Animal
```

```
{
```

```
public override void MakeSound()
```

```
{
```

```
Console.WriteLine("Woof!");
```

```
}
```

```
}
```

## نتیجه‌گیری

**Override** کردن متدها یکی از اصول اساسی در برنامه‌نویسی شیء‌گرا است که به برنامه‌نویسان امکان می‌دهد تا عملکرد متدها را در کلاس‌های فرزند به نیاز خود تغییر دهند. این امر از اهمیت بسیاری برخوردار است زیرا امکان پذیرش تغییرات در ساختار و عملکرد برنامه را فراهم می‌کند.