-

-

-

-

-

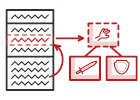
In Strategy pattern, **a class behavior or its algorithm can be changed at run time**. This type of design pattern comes under behavior pattern.

In Strategy pattern, we create objects which represent various strategies and a **context object whose behavior varies as per its strategy object**. **The strategy object changes the executing algorithm of the context object**.

**-**

**-**

**-**



# ****Strategy**** in C#

**Strategy** is a behavioral design pattern that turns a set of behaviors into objects and makes them interchangeable inside original context object.

**The original object, called context, holds a reference to a strategy object**. The **context delegates executing the behavior to the linked strategy object.** In order to change the way the context performs its work, other objects may replace the currently linked strategy object with another one.

**Complexity:** 1

**Popularity:**  3

**Usage examples:** The Strategy pattern is very common in C# code. It’s often used in various frameworks to provide users a way to change the behavior of a class without extending it.

**Identification:** Strategy pattern can be recognized by a method that lets a nested object do the actual work, **as well as a setter that allows replacing that object with a different one.**