**WZORCE PROJEKTOWE**

**STRATEGIA** – definiuje rodzinę algorytmów, pakuje je jako osobne klasy i powoduje, że są one w pełni wymienne. Zastosowanie tego wzorca pozwala na to, aby zmiany w implementacji algorytmów przetwarzania były całkowicie niezależne od strony klienta, który z nich korzysta.

**OBSERWATOR** – definiuje pomiędzy obiektami relację jeden-do-wielu w taki sposób, że kiedy wybrany obiekt zmienia swój stan, to wszystkie jego obiekty zależne zostają o tym powiadomione i automatycznie zaktualizowane.

**DEKORATOR** – pozwala na dynamiczne przydzielanie danemu obiektowi nowych zachowań. Dekoratory dają elastyczność podobną do tej, jaką daje dziedziczenie, oferując jednak w zamian znacznie rozszerzoną funkcjonalność.

**METODA FABRYKUJĄCA** – definiuje interfejs pozwalający na tworzenie obiektów, ale pozwala klasom podrzędnym decydować, jakiej klasy obiekt zostanie utworzony. Wzorzec Metoda Fabrykująca przekazuje odpowiedzialność za tworzenie obiektów do klas podrzędnych.

**FABRYKA ABSTRAKCYJNA** – dostarcza interfejs do tworzenia całych rodzin spokrewnionych lub zależnych od siebie obiektów bez konieczności określania ich klas rzeczywistych.

**SINGLETON** – zapewnia, że dana klasa będzie miała tylko i wyłącznie jedną instancję obiektu, i zapewnia globalny punkt dostępu do tej instancji.

(implementacja tego wzorca wykorzystuje **prywatny konstruktor** klasy, jak również **metody i zmienne statyczne**)