**В чем идея инкапсуляции? Абстракция, интерфейс и реализация. Средства защиты и доступа.**

**Инкапсуляция** – это свойство системы, позволяющее объединить данные и методы, работающие с ними, в классе и скрыть детали реализации от пользователя. Инкапсуляция неразрывно связана с понятием интерфейса класса. По сути, все то, что не входит в интерфейс, инкапсулируется в классе (сказ про сокрытие работы автомобиля).

Идея: (пример типа)

**Абстрагирование** – это способ выделить набор значимых характеристик объекта, исключая из рассмотрения незначимые. Тогда **абстракция** – это набор всех таких характеристик.

**Интерфейс** – это набор методов класса, доступных для использования другими классами. Интерфейсом класса будет являться набор всех его публичных методов в совокупности с набором публичных атрибутов. Таким образом интерфейс специфицирует класс четко определяя все возможные действия над ним (приборная панель автомобиля, которая позволяет методы увеличения скорости, торможения и так далее).

**Модификаторы доступа:**

1. **public** – доступ к типу или члену возможен из любого другого кода в той же сборке или другой сборке, ссылающейся на него
2. **private** – доступ к типу или к члену возможен только из кода в том же классе или структуре
3. **protected** – доступ к типу или члену возможен только из кода в том же классе либо в классе, производном от этого класса
4. **internal** – доступ к типу или члену возможен из любого кода в той же сборке, но не из другой сборки
5. **protected internal** – доступ к типу или к члену возможен из любого кода в той сборке, где он был объявлен, или из производного класса в другой сборке.
6. **private protected** – доступ к типу или члену возможен только из объявляющей сборки из кода в том же классе либо в типе, производном от этого класса