

# Лекция 1

ООП, классы, объекты, методы.

# Основные принципы ООП

## Инкапсуляция

Инкапсуляция — это свойство системы, позволяющее объединить данные и методы, работающие с ними в классе, и скрыть детали реализации от пользователя.

## Наследование

Наследование — это свойство системы, позволяющее описать новый класс на основе уже существующего с частично или полностью заимствующейся функциональностью. Класс, от которого производится наследование, называется базовым, родительским или суперклассом. Новый класс — потомком, наследником или производным классом.

## Полиморфизм

Полиморфизм — это свойство системы использовать объекты с одинаковым интерфейсом без информации о типе и внутренней структуре объекта.[1]

## Абстракция

Абстрагирование — это способ выделить набор значимых характеристик объекта, исключая из рассмотрения незначимые. Соответственно, абстракция — это набор всех таких характеристик.

# Класс и экземпляр

Класс - это шаблон, чертёж по которому можно создавать объекты реального мира - экземпляры классов.

Методы которые работают с классами, называют статическими.

# Примитивные типы в CLR

Примитив- ный тип	FCL тип	Совмести- мость с CLS	Описание
sbyte	System.Sbyte	Нет	8-разрядное значение со знаком
byte	System.Byte	Да	8-разрядное значение без знака
short	System.Int16	Да	16-разрядное значение со знаком
ushort	System.Uint16	Нет	16-разрядное значение без знака
int	System.Int32	Да	32-разрядное значение со знаком
uint	System.Uint32	Нет	32-разрядное значение без знака
long	System.Int64	Да	64-разрядное значение со знаком
ulong	System.Uint64	Нет	64-разрядное значение без знака
char	System.Char	Да	16-разрядный символ Unicode (char никогда не представляет 8-разрядное значение, как в неуправляемом коде на C++)
float	System.Single	Да	32-разрядное значение с плавающей точкой в стандарте IEEE
double	System.Double	Да	64-разрядное значение с плавающей точкой в стандарте IEEE
bool	System.Boolean	Да	Булево значение (true или false)
decimal	System.Decimal	Да	128-разрядное значение с плавающей точкой повышенной точности, часто используемое для финансовых расчетов, где недопустимы ошибки округления. Один разряд числа — это знак, в следующих 96 разрядах помещается само значение, следующие 8 разрядов — степень числа 10, на которое делится 96-разрядное число (может быть в диапазоне от 0 до 28). Остальные разряды не используются
string	System.String	Да	Массив символов
object	System.Object	Да	Базовый тип для всех типов

# Поле

\* Поле представляет собой немменяемое или изменяемое значение. Поле может быть статическим - тогда оно является частью типа. Поле может быть экземплярным (нестатическим) - тогда оно является частью объекта. Я настоятельно рекомендую делать поля закрытыми, чтобы внешний код не мог нарушить состояние типа или объекта.

# Метод

\* Метод представляет собой функцию, выполняющую операции, которые изменяют или запрашивают состояние типа (статический метод) или объекта (экземплярный метод). Методы обычно осуществляют чтение и запись полей типов или объектов.

# Константа

\* Константа - идентификатор, определяющий некую постоянную величину. Эти идентификаторы обычно используют, чтобы сделать код более читабельным, а также для удобства сопровождения и поддержки. Константы всегда связаны с типом, а не с экземпляром типа. Константы всегда статичны.

# Конструктор

- \* Конструктор экземпляров - метод, служащий для инициализации полей экземпляра при его создании.
- \* Конструктор типа - метод, используемый для инициализации статических полей типа.



# Свойство (Properties)

- \* Свойство представляет собой механизм, позволяющий применить простой синтаксис (напоминающий обращение к полям) для установки или получения части логического состояния типа или объекта, не нарушая это состояние.

# Остальное

- \* Событие. Статическое событие служит механизмом, позволяющим типу посылать уведомление статическому или экземпляру методу.
- \* Вложенные тип позволяет определять другие вложенные в него типы.
- \* Перегруженный оператор.
- \* Оператор преобразования.