В объектно-ориентированной технологии используется особый подход к разработке программ, основанный на использовании объектных моделей и нескольких базовых концепциях. К этим концепциям относятся **абстрагирование, инкапсуляция, полиморфизм, наследование.**

В программировании разработчики концентрируют свое внимание на существенных свойствах, необходимых для описания объекта, и на операциях, которые описывают его поведение. В этом и заключается ***абстрагирование***.

При абстрагировании выделяются те характеристики объекта, которые отличают его от всех других видов объектов и, таким образом, четко определяют его концептуальные границы с точки зрения наблюдателя.

***Инкапсуляция***. Этот термин характеризует сокрытие отдельных деталей внутреннего устройства класса от внешних по отношению к нему объектов или пользователей.

***Наследование*** – это механизм, который позволяет создавать новые классы на основе существующих, используя их структурные и поведенческие характеристики. Новые классы называют дочерними (производными или подклассами), а классы, на основе которых происходит наследование, – родительскими (базовыми или суперклассами). Кроме наследуемых свойств дочерние классы обладают дополнительными характеристиками, которые и отличают их от родительских.

***Полиморфизм*** – достаточно широкое понятие, в котором можно выделить следующие формы:

∙ Перегрузка методов;

∙ Перегрузка операций;

∙ Использование методов с одним и тем же именем в различных классах, включая виртуальные функции;

∙ Обобщенные классы (шаблоны классов).

**Перегруженные методы** - это методы с одинаковым названием, но с разным кол-вом параметров или с типом параметров.

**Перегрузка операций** - Удобнее использовать вместо метода Sum() при сложении двух матриц знак “+” для таких ситуаций используются перегрузка операций

Использование методов с одним и тем же именем в различных классах, включая **виртуальные функции:**

Например, классы «Учебная группа» и «Студент» могут содержать методы для печати информации о соответствующем объекте, которые имеют одинаковое имя, например, Печать(). При использовании метода Печать() будет выводиться информация о том объекте, для которого он вызван: при вызове метода для объекта класса «Учебная группа» будет печататься список студентов этой группы, а при вызове для объекта класса «Студент» – информация о студенте (ФИО, № зачетной книжки, № группы и др.). Это возможно за счет того, что **каждый объект знает, какому классу он принадлежит**, что позволяет вызвать метод именно этого класса.

Механизм использования ***виртуальных методов*** основывается на возможности хранения в переменной родительского класса объекта дочернего класса. По умолчанию выбор вызываемого метода осуществляется в соответствии с типом объекта, хранящегося в переменной. Если в родительском классе вызываемый метод был определен как виртуальный, а в дочернем классе он был переопределен, то будет вызываться последний.

***Обобщенные классы (шаблоны классов)*** определяют описание класса для обобщенного типа данных. При обращении к шаблону класса указывается конкретный тип данных (int, double или класс), который подставляется на место обобщенного типа. Таким образом, компилятор генерирует класс для этого конкретного типа. Подставляя разные типы данных, можно создать множество классов на основе шаблона, реализующих единый алгоритм обработки данных.