Абстрактные классы – классы, по которым невозможно создать экземпляр класса, необходимы для содержания общих методов и свойств иерархии классов. Абстрактный класс определяется ключевым словом abstract. Данный класс может содержать как абстрактные методы, так и простые методы. Иногда абстрактными становятся классы не имеющие ни одного абстрактного метода, это необходимо только для того чтобы невозможно было создать экземпляр данного класса. Абстрактные методы не содержат реализации, реализация обязательно должна присутствовать во всех классах, которые наследуются от абстрактного класса.

Интерфейс (Interface). ВСЕ методы интерфейса должны быть переписаны в классах, которые выполняет (implement) интерфейс. Переменных в интерфейсе не может быть, кроме констант. Уровень доступа в методах и константах интерфейсе всегда public, мы можем влиять только на сам интерфейс, т.е. если при определении интерфейса стоит ключевое слово public, то видно его везде в проекте, если нет, то только в текущем package. Существуют, так называемые, интерфейсы маркеры, которые не содержат ни констант, ни определения методов. Необходимы для пометки классов и осуществления некоторых действий, в случае если данный класс implement данный маркер интерфейс.

Сходство abstract class & interface – в наличие и там и там абстрактных методов, и все. Interface заставляет все классы, которые его выполняют осуществлять методы, которые в нем определены. Т.е. если в интерфейсе есть метод draw(), то класс обязан его осуществить( выполнить) и это может быть как класс Shape – нарисовать фигуру, так и класс Humen - нарисовать что-нибудь. В то же время абстрактный класс – это класс собирательный, т.е. класс, в который выделяются общие методы, свойства иерархии классов, которые его расширяют.