Vererbung:

Oberklasse vererbt alle ihre Eigenschaften und Methoden an die Unterklassen. Diese können weitere Methoden und Attribute haben, je nach Bedarf.

Abstrakte Klassen:

Abstrakte Klassen zeichnen sich dadurch aus, das in ihrer Beschreibung {abstract} folgt. Das verhindert, dass man konkrete Objekte dieser Klasse erzeugen kann. Es wäre z.B. relativ sinnfrei, abstrakte „Fortbewegungsmittel“ zu erzeugen, konkrete Fahrzeuge, wie ein Fahrrad, aber schon.

Subtyping:

Diese Eigenschaft sorgt dafür, dass die Unterklassen mit ihren Methoden auf alle weiteren Unterklassen ihrer Oberklasse zugreifen können. Es wäre sonst notwendig, für jede Art von Fahrzeug eine eigene Methode „fahren()“ zu erstellen. Mit dem Subtyping reicht dagegen eine Methode, die flexibel mit den Unterklassen interagiert. Beispielsweise sehe das dann so aus:

Fahren(Fortbewegungsmittel pFortbe). Das bedeutet nur, dass die Methode Fahren() ein Fortbewegungsmittel bekommt, ohne genau festzulegen welches (pFortbe). Das p steht hierbei nur für Parameter, also die übergebenen Werte. Beliebige andere Namen sind denkbar.

Typecasting:

Dies blockiert den Zugriff anderer Klassen auf die Methoden der Unterklasse (siehe S. 27, Beispiel mit dem Makler). So wird verhindert, das ungewünschte Interaktionen zwischen verschiedenen (Haupt)-Klassen entstehen.

Überschreiben:

Es kann passieren, dass die geerbten Methoden nicht die nötigen Eigenschaften für alle Unterklassen bereitstellen. Es ist daher möglich, deine Methode mit gleichem Namen innerhalb einer Unterklasse anzulegen, die jedoch andere Attribute benötigt oder andere Funktionen erfüllt.

Polymorphie

Durch das Überschreiben kann es unklar werden, wann welche Methode zum Einsatz kommen soll, da beide den gleichen Namen besitzen. Die ursprüngliche Methode der Hauptklasse, oder die der Unterklasse? Mit der Polymorphie wird beim Ausführen danach geschaut, welchen Typ das konkrete Objekt hat (also ob das Objekt aus einer Unterklasse mit überschriebener Methode kommt oder aus einer Klasse mit der Standardmethode).