**Nachdenkzettel Vererbung**

**Aufgabe 1:**

Dies hängt davon ab, ob die Methode in der Klasse X eine Implementierung bereitstellt. Es gibt also zwei Möglichkeiten:

Möglichkeit 1: Klasse X bietet eine Implementierung für die Methode:

Hier erbt Klasse B automatisch die Implementierung der neuen Methode und muss nicht angepasst werden.

Möglichkeit 2: Klasse X ist abstrakt bzw. bietet keine Implementierung für die neue Methode:

Hier muss die Methode in der abgeleiteten Klasse selbst implementiert werden.

**Aufgabe 2:**

Hier wird es zu einem Kompilierungsfehler kommen, da die Variable „x“ vom Typ „X“ ist, die Klasse „X“ die Methode „newMethodinB()“ gar nicht kennt, es werden nämlich nur auf Methoden zugegriffen, die in der Klasse „X“ definiert sind. Lösungen könnten sein:

1. Verwendung der Klasse B:

B b = new B();

b.newMethodinB();

1. Die Methode „newMethodinB()“ der Klasse X hinzufügen, die Klasse X abstrakt machen und dann in Klasse B implementieren. Allerdings könnte das auch zu einem Nachteil führen, da die implementierte Methode in X in jeder abgeleiteten Klasse ebenfalls implementiert werden müsste.
2. Das Casten von x auf den Typen B, um auf die Methode zuzugreifen.

((B)x).newMethodinB();

**Aufgabe 3:**

1. Bei der Annahme, dass String nicht final ist, wäre der Code soweit korrekt
2. Dadurch, dass Strings immutable sind, kann der String durch diese Aktion nicht verändert werden, da man einen neuen String für diese Aktion erstellen müsste. Würde man ebenfalls diese Eigenschaft von Strings entfernen, würde es quasi zu einer schon fast neuen Sprache kommen, da es einer der Grundprinzipien von Strings entfernen würde.
3. immutable