**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();

Je nach Anwendungsfall würde ich Methode 1 oder 2 präferieren

**Aufgabe 3:**

1. Bei der Annahme, dass String nicht final ist, wäre der Code so weit korrekt
2. Was passieren kann, ist dass der Code zwar erstmal akzeptiert wird, die Erstellung der Filenames bei manchen Betriebssystemen jedoch fehlschlagen könnte, da unterschiedliche Betriebssysteme unterschiedliche Anforderungen an Dateinamen haben (z.B bestimmte Zeichen, Länge etc.). Außerdem könnte die Übergabe an ein Betriebssystem in Form eines Strings ebenfalls verschlagen.
3. Das verstößt gegen das eigentliche Prinzip der Vererbung, was die vorherige Klasse spezifizieren oder erweitern soll und nicht eine Veränderung hervorruft.