1. **Software/Web Engineering:**
2. Geben Sie eine allgemeine Definition wieder und listen Sie die Bestandteile des Software Engineering auf.
3. Welche Besonderheiten gibt es im Bereich Web Engineering im Vergleich zu herkömmlichen Software Engineering (Ziele, Personen, Programmiersprachen)?
4. Was versteht man unter Continuous Delivery, was benötigt man um dies umzusetzen und welche Vorteile erwartet man sich davon?
5. **GIT:**
6. Nennen Sie einige Vorteile von GIT als Versionskontrollsystem.
7. Was verstehen Sie unter nicht linearer Entwicklung?
8. Welche GIT Kommandos kennen Sie und was kann man damit machen (min. 3 Kommandos, max. 5)?
9. Was ist ein Merge Konflikt und wie geht man damit um?
10. Erklären Sie den 3-Wege-Merge anhand einer Darstellung.
11. **Refactoring:**
12. Welchen Autor würden Sie nachschlagen wenn Sie mehr über Refactoring lernen möchten?
13. Zählen Sie positive und negative Punkte zu Refactoring im Allgemeinen auf.
14. Was ist die ultimative Voraussetzung um ein Refactoring durchzuführen?
15. Suchen Sie sich ein Refactoring aus, erklären Sie dieses anhand der Struktur die Fowler in seinem „Catalog of Refactorings“ gewählt hat.
16. Geben Sie ein Codebeispiel für das von Ihnen gewählte Refactoring. Schreiben Sie auf einem Blatt Papier - links den ursprünglichen Code und rechts das Refactoring (Programmiersprache nach Wahl, C-style, Ruby oder Pseudocode)