

# Lernnachweis zu Kompetenz C1G

## Wesentliche Refactoring-Techniken

Die Verbesserung der Lesbarkeit und Verständlichkeit von Code ist ein entscheidender Aspekt der Softwareentwicklung. Einige wesentliche Refactoring-Techniken, die diesen Zweck unterstützen:

### 1. Extraktion von Methoden:

```
def complex_algorithm():
    helper_function_1()
    helper_function_2()

def helper_function_1():
    # ... Teil des ursprünglichen Codes ...

def helper_function_2():
    # ... Teil des ursprünglichen Codes ...
```

### 2. Aussagekräftige Benennung:

Vorher:

```
x = process_data(a, b)
```

Nachher:

```
result = calculate_average_score(math_scores, english_scores)
```

### 3. Entfernung von Redundanz:

Vorher:

```
if condition == True:
```

Nachher:

```
if condition:
```

Diese wenigen, aber wesentlichen Refactoring-Techniken fördern die Lesbarkeit und Wartbarkeit des Codes erheblich. Die Anwendung solcher Prinzipien verbessert nicht nur die Codequalität, sondern erleichtert auch die Zusammenarbeit im Team und die Pflege des Codes über die Zeit.