

Ziel des Codes

Der Service `AmortizationService` berechnet einen **Tilgungsplan** (Amortization Schedule) und eine **Zusammenfassung** für einen Kredit anhand gegebener Kreditdaten wie Darlehensbetrag, Zinssatz, anfänglicher Tilgung und Zinsbindungsfrist.

Wichtige Eingabedaten (`LoanData`)

- `loanAmount` : Gesamtdarlehensbetrag (z. B. 100.000 €)
- `interestRate` : jährlicher Zinssatz in Prozent (z. B. 3 %)
- `initialRepayment` : anfängliche Tilgung in Prozent jährlich (z. B. 2 %)
- `interestFixation` : Zinsbindungsfrist in Jahren (z. B. 10)

Schrittweise Erklärung der Methode `calculateAmortizationPlan()`

1. Validierung der Eingabedaten

```
if (!this.isValidLoanData(loanData)) {  
    throw new InvalidLoanDataError(...);  
}
```

Stellt sicher, dass alle Zahlen positiv (außer Zinssatz = 0 erlaubt) und vorhanden sind.

2. Berechnung der Monatsrate

```
const monthlyInterestRate = interestRate / 100 / 12;  
const numberOfPayments = interestFixation * 12;  
const monthlyPayment = this.calculateMonthlyPayment(...);
```

Monatlicher Zinssatz

Wird aus dem Jahreszins berechnet, geteilt durch 12 Monate.

Anzahl der Raten

Multipliziert die Jahre der Zinsbindung mit 12 (Monate pro Jahr).

Monatsrate (calculateMonthlyPayment)

```
annualPayment = loanAmount * (interestRate / 100 + initialRepayment / 100)
monthlyPayment = annualPayment / 12
```

✅ Warum so?

- Die anfängliche jährliche Gesamtbelastung = Zinsen + anfängliche Tilgung (beide in Prozent vom Darlehen).
- Diese Jahresrate wird auf 12 Monate verteilt.

3. Zinsprüfung für den ersten Monat

```
const firstMonthInterest = loanAmount * monthlyInterestRate;

if (monthlyPayment <= firstMonthInterest + ZERO_THRESHOLD) {
  throw new InsufficientRateError(...);
}
```

Warum? Die Rate muss mindestens die Zinsen decken. Sonst findet keine Tilgung statt → die Schulden wachsen oder bleiben ewig gleich.

4. Erstellung des Tilgungsplans (Array)

Auszahlungseintrag

```
createDisbursementEntry(loanAmount, currentDate)
```

Eintrag mit:

- Negativer Restschuld (–Darlehen)
- Keine Zinsen oder Tilgung
- Negativer Cashflow (Auszahlung)

Schleife: Monatliche Zahlungen

```
for (let i = 0; i < numberOfPayments; i++) {
  const previousBalance = lastEntry.remainingDebt;
  if (nearZero(previousBalance)) {
    push ZeroEntry
  } else {
    push calculateNextPaymentEntry(...)
  }
}
```

```
}  
}
```

Jede Iteration erstellt einen neuen Monats-Eintrag:

- Basiert auf dem vorherigen Stand der Restschuld
- Berechnet Zinsen, Tilgung und Restschuld neu



Detailliert: calculateNextPaymentEntry()

Eingang:

- `previousBalance` (negativ oder 0)
- `monthlyPayment` (vorher berechnet)
- `monthlyInterestRate`
- Datum

1. Zinsberechnung

```
interest = -previousBalance * monthlyInterestRate
```

Da `previousBalance` negativ ist, ergibt sich ein positiver Zinsbetrag.

2. Tilgung (Kapitalanteil)

```
principalPayment = monthlyPayment - interest;
```

Was von der Monatsrate nach Abzug der Zinsen übrig bleibt, geht in die Tilgung.

3. Sonderfall: Letzte Zahlung

```
if (principalPayment >= restschuld - Toleranz) {  
    principalPayment = Restschuld;  
    actualPayment = interest + principalPayment;  
}
```

Wenn der verbleibende Betrag fast vollständig gedeckt ist, wird die Zahlung angepasst (damit nichts überzahlt wird).

4. Neue Restschuld

```
newBalance = previousBalance + principalPayment;
```

Die Tilgung wird vom vorherigen Stand abgezogen (bzw. aufs Konto des Schuldners angerechnet).



calculateSummaryData()

Erstellt ein Summenobjekt für:

- Gesamtzinszahlungen
- Gesamt-Tilgungen
- Übrig bleibende Schuld am Ende der Laufzeit

Berechnung:

```
totalInterestPaid = Summe aller entry.interest  
totalRepaymentPaid = Summe aller entry.repayment  
remainingDebt = letzte remainingDebt (gerundet)
```



Rundung und Toleranzen

```
roundToTwoDecimals(value)  
ZERO_THRESHOLD = 0.005
```

Damit Rundungsfehler (z. B. 0.0049 €) die Logik nicht stören, wird mit einem kleinen Schwellenwert gearbeitet, und alles auf 2 Dezimalstellen gerundet.



Zusammenfassung der Logik

Schritt	Beschreibung	Warum
Validierung	Prüft auf gültige Eingaben	Fehlervermeidung
Monatszins & Raten	Berechnung auf Basis Jahreswerte	Grundlage für Plan
Prüfung: Rate deckt Zinsen	Vermeidet negative Tilgung	Realistische Rückzahlung
Initialer Eintrag	Auszahlung dokumentieren	Startpunkt

Schritt	Beschreibung	Warum
Monatliche Schleife	Tilgung & Zinsen Monat für Monat	Plan über Laufzeit
Letzte Rate anpassen	Genauigkeit & kein Überzahlen	Präzision
Zusammenfassung	Übersicht über Zins, Tilgung, Rest	Nutzerinformation