

Aufgabe 1 *Implementierung der Klasse Konto*

Die Klasse **Konto** modelliert ein einfaches Bankkonto. Die Eigenschaften und Methoden sind durch folgendes UML-Diagramm gegeben:

| Konto |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| - iban: String - ueberziehungsRahmen: int - saldo: int |
| + Konto() + Konto(iban: String, ueberziehungsRahmen: int) + getIban(): String + setIban(iban: String): void + getUeberziehungsRahmen(): int + setUeberziehungsRahmen(ueberziehungsRahmen: int): void + einzahlen(betrag: int): void + abbuchen(betrag: int): boolean + kontostand(): int + berechneZinsen(zinssatz: double): void + toString(): String + print(): void |

Das Attribut **saldo** gibt den aktuellen Kontostand in Euro an und wird intern verwaltet (also ohne Getter/Setter). Die **iban** und der **ueberziehungsRahmen** werden über den Konstruktor gesetzt und können über Getter und Setter abgefragt bzw. verändert werden. Der Standardwert für den **ueberziehungsRahmen** soll 1000 sein. Im parameterlosen Konstruktor soll ein IBAN zufällig generiert werden. Für den IBAN gilt allgemein: Er besteht aus 20 Zeichen, wobei die ersten beiden Zeichen Großbuchstaben sind, die verbleibenden Zeichen sind Ziffern.

Die Methode **einzahlen(int betrag)** erhöht den Kontostand um den angegebenen Betrag. Die Methode **abbuchen(int betrag)** versucht, den angegebenen Betrag vom Konto abzubuchen. Dabei gilt: eine Abbuchung darf nur durchgeführt werden, wenn das Konto dadurch nicht über den Überziehungsrahmen hinaus belastet wird. Wird die Abbuchung erfolgreich durchgeführt, so gibt die Methode **true** zurück, andernfalls **false**.

Die Methode **berechneZinsen(double zinssatz)** addiert zum aktuellen Kontostand die Jahreszinsen, die sich aus dem angegebenen Zinssatz ergeben. Das Ergebnis der Berechnung wird auf eine ganze Zahl gerundet und dem Kontostand gutgeschrieben.¹

Die **toString**- bzw. **print**-Methoden sollen die Eigenschaften des Kontos in folgendem Format ausgeben:

```
Konto AT61 1904 3002 3457 3201, Saldo: 2500 €, Überziehungsrahmen: 1000 €  
Konto AT61 1904 3002 3457 3201, Saldo: -800 €, Überziehungsrahmen: 1000 €  
überschritten!
```

Prüfen Sie alle Parameter auf Gültigkeit und geben Sie entsprechende Fehlermeldungen aus. Wenn keine expliziten Gültigkeitsbereiche angegeben sind, treffen Sie sinnvolle Annahmen! Schreiben Sie ausführliche JUnit-Tests für diese Klasse.

¹ *Hinweis:* Die tatsächliche Zinsberechnung in Banken ist deutlich komplexer (z. B. mit Tagesgenauigkeit, unterschiedlichen Zinssätzen für Guthaben und Schulden etc.).

Lösung zur Generierung des IBANs:

```
1 private String generateIban() {
2     Random rnd = new Random();
3     long ibanLong = 1000_0000_0000_0000L + Math.abs(rnd.nextLong());
4     String ibanStr = "" + ibanLong;
5     ibanStr = ibanStr.substring(0, 18);
6
7     char c1 = (char)(65 + rnd.nextInt(25));
8     char c2 = (char)(65 + rnd.nextInt(25));
9     ibanStr = "" + c1 + c2 + ibanStr;
10    return ibanStr;
11 }
```