

## Ausnahmen implementieren (25 Punkte)

Bei einem Autoverleih gelten die folgenden Preise:

Bei einer Mietdauer von weniger als 14 Tagen werden pro Tag 20 Euro berechnet.

Zusätzlich wird bis einschließlich 200km eine Pauschale von 40 Euro fällig.

Bei mehr als 200km werden zusätzlich zur Pauschale pro weiterem Kilometer 15 Cent berechnet.

Ab 14 Tagen Mietdauer wird eine Pauschale von 300 Euro fällig und 12 Cent pro gefahrenem Kilometer.

Implementieren Sie in der Datei [MeinMietpreisBerechner.java](#) nun die Methode, die die Mietpreisberechnung durchführt. Dabei soll im Falle einer nicht positiven Eingabe für Tage eine `UngueltigeTagesanzahlAusnahme` und im Falle einer negativen Kilometeranzahl eine `UngueltigeKilometerangabeAusnahme` geworfen werden. Beachten Sie dabei, dass die beiden Klassen `UngueltigeTagesanzahlAusnahme` und `UngueltigeKilometerangabeAusnahme` nicht als `public` deklariert werden dürfen.

Meine eingereichte Lösung.

Alle Tests des Testfalls wurden erfolgreich bestanden.

Was nicht im Text erwähnt ist: der Testfall erwartet die Berechnung in Cent.

**Keine Garantie** auf Richtigkeit oder Vollständigkeit.

Bitte ändert/modifiziert die Kommentare, wenn ihr die Lösung übernehmen wollt (macht's vielleicht auch ohne Konstanten oder ändert deren Namen).

```
public class MeinMietpreisBerechner {  
  
    public long berechneMietpreis(int tage, int km) throws  
        UngueltigeTagesanzahlAusnahme, UngueltigeKilometerangabeAusnahme {  
  
        //hier beginnt der Code, der Eingereicht werden muss!!!  
  
        //lokale Variablen  
        long endpreis = 0;  
        //Konstanten zur Berechnung - Aufgrund der Einreichungsform nicht als  
        Klassenattribute möglich  
        final int TAGESRATE = 2000;  
        final int PAUSCHALE_LANGZEIT = 30000;  
        final int KILOMETERPAUSCHALE = 4000;  
        final int KM_PREIS_KURZZEIT = 15;
```

```

final int KM_PREIS_LANGZEIT = 12;

//Block zur Ausnahmebehandlung
if(tage <= 0) {
    throw new UngueltigeTagesanzahlAusnahme("Ungültige Anzahl an
        Tagen");
    //Fehlermeldung bei 0 oder negativer Anzahl an Tagen
}
if(km < 0) {
    throw new UngueltigeKilometerangabeAusnahme("Ungültige Anzahl
        an Kilometern");
    //Fehlermeldung bei negativer Anzahl an Km
}
//Ende Ausnahmebehandlung
if(tage < 14) {
    endpreis = tage * TAGESRATE;
    //Berechnung endpreis bei Nutzung unter 14 Tage
    if (km <= 200) {
        endpreis += KILOMETERPAUSCHALE;
        //Berechnung endpreis bei Nutzung unter 14 Tage und
        unter 200 km
    }else {
        endpreis += KILOMETERPAUSCHALE + (km - 200) *
            KM_PREIS_KURZZEIT;
        //Berechnung endpreis bei Nutzung unter 14 Tage und
        über 200 km
    }
}
else {
    endpreis = PAUSCHALE_LANGZEIT + km * KM_PREIS_LANGZEIT;
    //Berechnung endpreis bei Nutzung länger als 14 Tage
}
return endpreis;

```

//hier endet der Code, der Eingereicht werden muss!!!

}

}

```

class UngueltigeTagesanzahlAusnahme extends Exception{

```

```

    UngueltigeTagesanzahlAusnahme(String s){
        super(s);
    }

```

}

```

class UngueltigeKilometerangabeAusnahme extends Exception{

```

```

    UngueltigeKilometerangabeAusnahme(String s){
        super(s);
    }

```

}