

Hinweis: Es sind alle Felder auszufüllen! Abgabe der Übungsblätter immer **mittwochs** (Ausnahme wenn Feiertag: donnerstags) bis **spätestens 12:00 Uhr** in die entsprechend gekennzeichneten Briefkästen der Veranstaltung im Erdgeschoss des Instituts für Informatik (Gebäude N). Zuwiderhandlung wird mit Strafe geahndet! (Punktabzug)

Übungsblatt	
-------------	--

(hier die Nummer des bearbeiteten **Übungsblatts** eintragen)

	Übung 01 (1057 N) Montag 08:15 - 09:45 Uhr (Isabell Rücker)
	Übung 02 (1056 N) Montag 14:00 - 15:30 Uhr (Henning Cui)
	Übung 03 (1057 N) Montag 15:45 - 17:15 Uhr (Josef Kircher)
	Übung 04 (1054 N) Montag 17:30 - 19:00 Uhr (Mosaab Slimani)
	Übung 05 (1057 N) Montag 17:30 - 19:00 Uhr (David Hacker)
	Übung 06 (1055 N) Dienstag 12:15 - 13:45 Uhr (André Schweiger)
X	Übung 07 (1054 N) Dienstag 17:30 - 19:00 Uhr (Benjamin Sertolli)
	Übung 08 (1057 N) Dienstag 17:30 - 19:00 Uhr (Dat Le Thanh)
	Übung 09 (1054 N) Mittwoch 08:15 - 09:45 Uhr (Erik Pallas)
	Übung 10 (1055 N) Mittwoch 08:15 - 09:45 Uhr (Moritz Feldmann)
	Übung 11 (1054 N) Mittwoch 10:00 - 11:30 Uhr (Denise Böhm)
	Übung 12 (1056 N) Donnerstag 08:15 - 09:45 Uhr (Florian Magg)
	Übung 13 (1054 N) Donnerstag 15:45 - 17:15 Uhr (Marvin Drexelius)
	Übung 14 (1054 N) Donnerstag 17:30 - 19:00 Uhr (Patrick Eckert)
	Übung 15 (1057 N) Donnerstag 17:30 - 19:00 Uhr (Alexander Szöke)
	Übung 16 (1057 N) Freitag 08:15 - 09:45 Uhr (Philipp Braml)
	Übung 17 (1054 N) Freitag 10:00 - 11:30 Uhr (Elisabeth Korndörfer)
	Übung 18 (1054 N) Freitag 12:15 - 13:45 Uhr (Philipp Häusele)
	Übung 19 (1056 N) Freitag 12:15 - 13:45 Uhr (Maximilian Demmler)
	Übung 20 (1054 N) Freitag 14:00 - 15:30 Uhr (Florian Straßer)

(hier die eingeteilte **Übungsgruppe** ankreuzen)

Teamnummer	6
------------	---

(hier die Nummer des eingeteilten **Teams** eintragen)

Tarik Selimovic
Anton Lydike
Dominic Cesnak

(hier die **Vor- und Nachnamen** aller Teammitglieder eintragen)

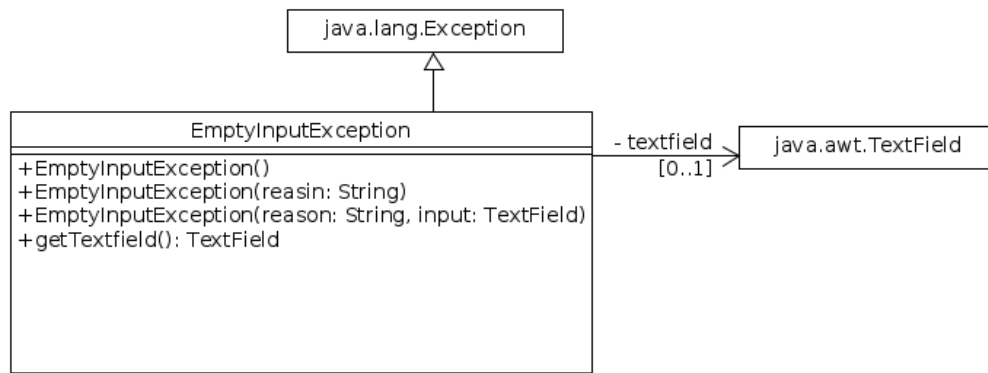
Aufgabe		
Aufgabe		
Aufgabe		
Aufgabe		
Gesamt		

(vom Tutor auszufüllen)

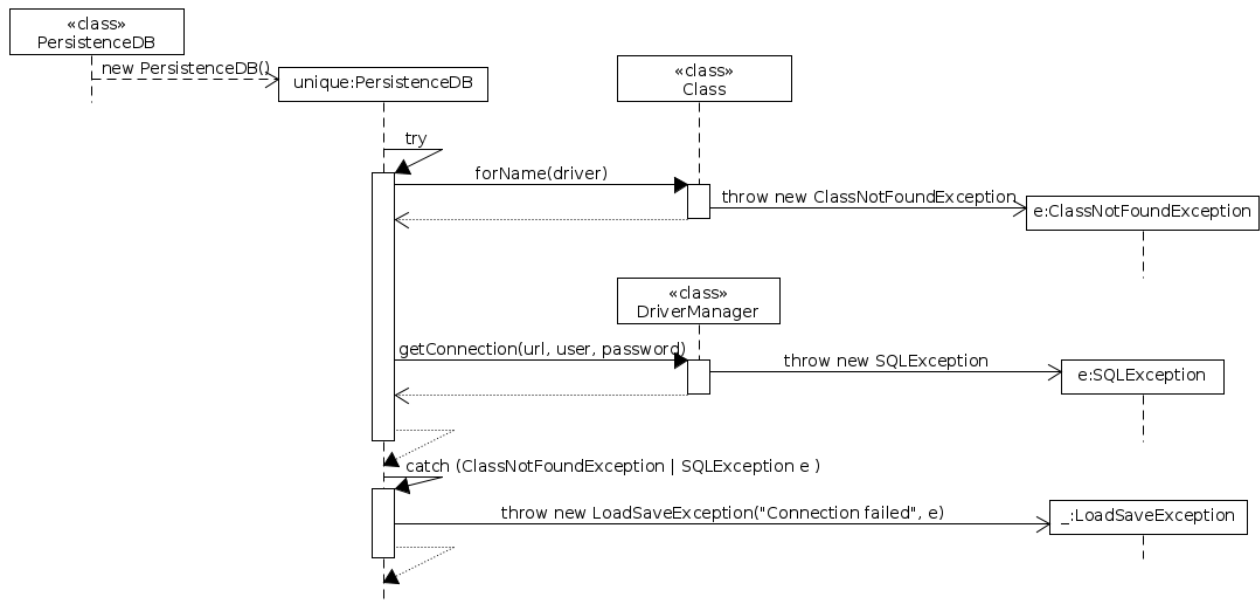
Übungsblatt 10

37)

a)



b)



40)

Datenbank)

```

package aufgabe40;

import java.sql.*;

public class Datenbank {
    private final int DBOP_DELETE = 1;
    private final int DBOP_INSERT = 0;
    private Connection con;

    public void add(Buch b) throws LoadSaveException {
        dbop(b, DBOP_INSERT);
    }

    public void delete (Buch b) throws LoadSaveException {
        dbop(b, DBOP_DELETE);
    }

    private void dbop(Buch b, int op) throws LoadSaveException {
        PreparedStatement s;
        try {
            con = this.connect();
            switch (op) {
                case DBOP_DELETE:
                    s = con.prepareStatement("REMOVE FROM buch WHERE isbn = ?");
                    s.setString(1, b.getISBN());
                    break;
                case DBOP_INSERT:
                    s = con.prepareStatement("INSERT INTO buch (`isbn`, `title`) VALUES (?, ?)");
                    s.setString(1, b.getISBN());
                    s.setString(2, b.getTitel());
                    break;
                default:
                    throw new LoadSaveException("What?", new WhatTheFuckJustHappenedException());
            }
            s.executeUpdate();
        } catch (SQLException e) {
            throw new LoadSaveException("It worked on my machine", e);
        }

        try {
            if (s != null) s.close();
        } catch (Exception e) {}

        try {
            if (con != null) con.close();
        } catch (Exception e) {}
    }

    // stub
    private Connection connect() {return null;}
}

// more stubs
class Buch {
    public String getTitel() {return "JavaScript: The Definitive Guide";}
    public String getISBN() {return "9781495333347";}
}

// damn, thats a lot of stubs
class LoadSaveException extends Exception {
    public LoadSaveException (String msg, Throwable t) {super(msg, t);}
}

// quite *stub*born... hehe
class WhatTheFuckJustHappenedException extends Exception {}

```

