

# INFI-DB 3AHWII

## WS+SS 2025

Sabo RUBNER - RubnS - sabo.rubner@htlinn.ac.at

24. Februar 2026

# Übersicht

## INFI DB

Folien github

Test-Termine

Git-Repositories

# Folien github Link

- ▶ Folien pull [https://github.com/htlinnrain/infi\\_3klasse](https://github.com/htlinnrain/infi_3klasse)
- ▶ Discord Server <https://discord.gg/hrjZSeHx5H>

# Test-Termine

...bitte erinnern falls nicht in Webuntis eingetragen

- ▶ T1 - 2025.11.12
- ▶ T2 - 2026.01.21
- ▶ T3 -
- ▶ T4 -

# Git-Repositories Schüler

- ▶ Andre <https://github.com/K-Beep-cmyk/Andre-INFI.git>
- ▶ Dvoryanov <https://github.com/nyslock/dvoryanov-infi>
- ▶ Kühleitner <https://github.com/Alex4848684/Infi>
- ▶ Larcher [https://github.com/johalar/INFI\\_3.Klasse](https://github.com/johalar/INFI_3.Klasse)
- ▶ Leitner [https://github.com/FinnLeitner/infi\\_3.Klasse](https://github.com/FinnLeitner/infi_3.Klasse)
- ▶ Muratgül <https://github.com/MRG0571/Infi>
- ▶ Özensoy [https://github.com/abdi38-1/INFI\\_3AHWII](https://github.com/abdi38-1/INFI_3AHWII)
- ▶ Öztürk [https://github.com/C4H10F02P/INFI\\_Malik\\_Oeztuerk](https://github.com/C4H10F02P/INFI_Malik_Oeztuerk)
- ▶ Putz <https://github.com/mpu703823-sudo/Infi>
- ▶ Sentürk [https://github.com/dogukansentrk/Sentuerk\\_INFI.git](https://github.com/dogukansentrk/Sentuerk_INFI.git)
- ▶ Zitturi <https://github.com/Gabulan54635/Informatik>

# Ablauf 2025.09.17

- ▶ Vorstellung Schüler
- ▶ git account registrieren, repository erzeugen
- ▶ was ist eine Datenbank?
- ▶ was ist ein Algorithmus?
- ▶ was ist SQL?
- ▶ was sind SQLite, MySQL, Oracle, DB2?
- ▶ SQLite wird z.B. bei Android/IOS/Browser verwendet
- ▶ wie sucht man etwas in einer Textdatei?
- ▶ wie sucht prinzipiell eine DB (Datenbank) etwas? Bäume!
- ▶ Array und ArrayList Vergleich
- ▶ Datenstrukturen allgemein (Array, Liste)
- ▶ Datentypen Programmiersprachen / SQLite

# Ablauf 2025.09.17

- ▶ Bisheriger Kenntnisstand?
- ▶ Java + SQLite
- ▶ Später Java + MySQL
- ▶ Was ist SQLite z.B. auf <http://de.wikipedia.org/wiki/SQLite>
- ▶ Warum SQLite:
  - ▶ Einfache Installation
  - ▶ Kein Prozess im Hintergrund nötig
  - ▶ Alle Datenbank-Dateien in einer Datei
- ▶ Arbeitsauftrag Wikipedia SQLite lesen
- ▶ Arbeitsauftrag Installations SQLite+Java
- ▶ <https://www.sqlitetutorial.net/download-install-sqlite/>
- ▶ Arbeitsauftrag funktionierende SQLite installation vorzeigen

# Ablauf 2025.09.24

- ▶ sqlite3.exe ausprobieren, CREATE, INSERT, SELECT ausprobieren
- ▶ für kleinere Anwendung völlig ausreichender Funktionsumfang
- ▶ Übersicht:  
[http://www.tutorialspoint.com/sqlite/sqlite\\_overview.htm](http://www.tutorialspoint.com/sqlite/sqlite_overview.htm)
- ▶ Installation:  
[http://www.tutorialspoint.com/sqlite/sqlite\\_installation.htm](http://www.tutorialspoint.com/sqlite/sqlite_installation.htm)
- ▶ Tutorial: [http://www.tutorialspoint.com/sqlite/sqlite\\_java.htm](http://www.tutorialspoint.com/sqlite/sqlite_java.htm)

# Ablauf 2025.10.01 und 2025.10.08

- ▶ Tabellen erzeugen:  
[https://www.tutorialspoint.com/sqlite/sqlite\\_create\\_table.htm](https://www.tutorialspoint.com/sqlite/sqlite_create_table.htm)
- ▶ Tabellen löschen:  
[https://www.tutorialspoint.com/sqlite/sqlite\\_drop\\_table.htm](https://www.tutorialspoint.com/sqlite/sqlite_drop_table.htm)
- ▶ Daten einfügen:  
[https://www.tutorialspoint.com/sqlite/sqlite\\_insert\\_query.htm](https://www.tutorialspoint.com/sqlite/sqlite_insert_query.htm)
- ▶ Daten auslesen:  
[https://www.tutorialspoint.com/sqlite/sqlite\\_select\\_query.htm](https://www.tutorialspoint.com/sqlite/sqlite_select_query.htm)
- ▶ where Bedingung:  
[https://www.tutorialspoint.com/sqlite/sqlite\\_where\\_clause.htm](https://www.tutorialspoint.com/sqlite/sqlite_where_clause.htm)
- ▶ Daten aktualisieren:  
[https://www.tutorialspoint.com/sqlite/sqlite\\_update\\_query.htm](https://www.tutorialspoint.com/sqlite/sqlite_update_query.htm)
- ▶ Daten löschen:  
[https://www.tutorialspoint.com/sqlite/sqlite\\_delete\\_query.htm](https://www.tutorialspoint.com/sqlite/sqlite_delete_query.htm)
- ▶ AND und OR: [https://www.tutorialspoint.com/sqlite/sqlite\\_and\\_or\\_clauses.htm](https://www.tutorialspoint.com/sqlite/sqlite_and_or_clauses.htm)
- ▶ Like:  
[https://www.tutorialspoint.com/sqlite/sqlite\\_like\\_clause.htm](https://www.tutorialspoint.com/sqlite/sqlite_like_clause.htm)
- ▶ ...durcharbeiten und in Java während der Stunde, Rest als HÜ einbauen



# Ablauf 2025.10.15

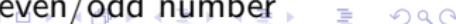
- ▶ Wiederholung und üben SQLITE + Java
- ▶ Datentypen SQLITE besprechen  
<https://www.sqlite.org/datatype3.html>

# Ablauf 2025.10.22 - HÜ Ferien

- ▶ Primary key <https://www.sqlitetutorial.net/sqlite-primary-key/>
- ▶ Autoincrement [https://www.tutorialspoint.com/sqlite/sqlite\\_using\\_autoincrement.htm](https://www.tutorialspoint.com/sqlite/sqlite_using_autoincrement.htm), <https://www.sqlite.org/autoinc.html>
- ▶ Tabelle anlegen, id=autoincrement, value=zufallszahl(1-10), value2=value%2 (value\_MODULO\_2)
- ▶ 20 Zeilen einfügen
- ▶ Anzahl gerader und ungerader values mit JAVA im Terminal ausgeben

# Ablauf 2025.11.05

- ▶ HÜ ansehen
- ▶ Unique
  - <https://www.sqlitetutorial.net/sqlite-unique-constraint/>
- ▶ Operatoren:
  - [https://www.tutorialspoint.com/sqlite/sqlite\\_operators.htm](https://www.tutorialspoint.com/sqlite/sqlite_operators.htm)
- ▶ Expressions:
  - [https://www.tutorialspoint.com/sqlite/sqlite\\_expressions.htm](https://www.tutorialspoint.com/sqlite/sqlite_expressions.htm),
  - [https://www.sqlite.org/lang\\_expr.html](https://www.sqlite.org/lang_expr.html)
- ▶ Glob:
  - [https://www.tutorialspoint.com/sqlite/sqlite\\_glob\\_clause.htm](https://www.tutorialspoint.com/sqlite/sqlite_glob_clause.htm),
  - <https://www.sqlitetutorial.net/sqlite-glob/>
- ▶ Unterschied zwischen Glob und Like?
- ▶ Limit:
  - [https://www.tutorialspoint.com/sqlite/sqlite\\_limit\\_clause.htm](https://www.tutorialspoint.com/sqlite/sqlite_limit_clause.htm)
- ▶ Java Args besprechen [https://wiki.byte-welt.net/wiki/4\\_Grundrechenarten\\_auf\\_der\\_Kommandozeile](https://wiki.byte-welt.net/wiki/4_Grundrechenarten_auf_der_Kommandozeile)
- ▶ [https://wiki.byte-welt.net/wiki/Java-Programm\\_mit\\_Argumenten\\_starten](https://wiki.byte-welt.net/wiki/Java-Programm_mit_Argumenten_starten)
- ▶ löschenmethode einbauen, beim Starten abfrage ob even/odd number



# Ablauf 2025.11.12

- ▶ überarbeitete/verbesserte HÜ Aufgabe ansehen
- ▶ Foreign key besprechen  
<https://www.sqlitetutorial.net/sqlite-foreign-key/>
- ▶ HÜ: beliebiges Beispiel zur Demonstration von Foreign keys in sqlite vorbereiten

# Ablauf 2025.11.19

- ▶ Testrückgabe
- ▶ Foreign key wiederholen, HÜ/Beispiel ansehen
- ▶ prepared statements  
[https://en.wikipedia.org/wiki/Prepared\\_statement](https://en.wikipedia.org/wiki/Prepared_statement)
- ▶ [https://imgs.xkcd.com/comics/exploits\\_of\\_a\\_mom.png](https://imgs.xkcd.com/comics/exploits_of_a_mom.png)
- ▶ Vorteile: Sicherheit und Geschwindigkeit
- ▶ Sql Injection: <https://www.hackedu.com/blog/how-to-prevent-sql-injection-vulnerabilities-how-prepared-statements-work>
- ▶ Beispiel: <https://www.securai.de/veroeffentlichungen/blog/sql-injection-beispiel-erklaerung-vermeidung/>
- ▶ von SQLite auf Mysql wechseln, vorherige Aufgabe mit even/odd umschreiben auf Mysql
- ▶ <https://www.vogella.com/tutorials/MySQLJava/article.html>

# Ablauf 2025.12.03

- ▶ mysql version ansehen der odd/even Aufgabe ansehen
- ▶ foreign keys nochmal besprechen
- ▶ Kunden-Artikel-Bestellungsbeispiel Abgabe/besprechen
- ▶ programmiere folgende Aufgabe mit JAVA+MySQL
  - ▶ erzeuge eine Kundentabelle mit (id, name, email)
  - ▶ erzeuge eine Artikeltablelle mit (id, bezeichnung, preis)
  - ▶ autoincrement verwenden
  - ▶ erzeuge alle notwendigen Methoden um die Tabellen anzulegen und zu befüllen
  - ▶ trenne Kunden- und Artikelbereiche in verschiedene Klassen
  - ▶ erzeuge eine Bestelltablelle mit (kundenID, artikelID, anzahl) (verwende Foreign Keys!!)
  - ▶ erzeuge alle notwendigen Methoden um einen Artikel zu bestellen
  - ▶ erzeuge eine Methode um die Bestellung von einem Kunden anzuzeigen
  - ▶ wenn Programm neu startet, bestehende DB mit alten Daten benutzen

# Ablauf 2025.12.10

- ▶ HÜ besprechen
- ▶ Erweiterungsaufgabe besprechen
  - ▶ Kunden-Artikel-Bestellungsbeispiel Erweiterungen (Update/Deletes einbauen)
  - ▶ Lagerhaltung einbauen
  - ▶ Interessante Abfragen überlegen
  - ▶ Terminal Bedienung einbauen

# Ablauf 2025.12.17

- ▶ Weihnachtspullover anziehen
- ▶ Kekse mitnehmen erlaubt :-)
- ▶ HÜ ansehen
- ▶ DATE, DATETIME besprechen
  - ▶ <https://dev.mysql.com/doc/refman/8.0/en/date-and-time-types.html>
- ▶ Kunden-Artikel-Bestellungsbeispiel erweitern mit DATE und DATETIME,  
z.B. wann wurde ein Artikel angelegt, wann wurde eine Bestellung aufgegeben

# Ablauf 2026.01.07

- ▶ Datums HÜ ansehen
- ▶ Files lesen+schreiben-; import/export DB besprechen
- ▶ JSON
- ▶ einlesen <https://www.geeksforgeeks.org/different-ways-reading-text-file-java/>
- ▶ schreiben <https://www.geeksforgeeks.org/file-handling-java-using-filewriter-filereader/>
- ▶ JSON encode, decode  
[https://www.tutorialspoint.com/json/json\\_java\\_example.htm](https://www.tutorialspoint.com/json/json_java_example.htm)
- ▶ JSON einlesen <https://www.tutorialspoint.com/how-can-we-read-a-json-file-in-java>
- ▶ JSON schreiben <https://www.tutorialspoint.com/how-to-write-create-a-json-file-using-java>
- ▶ Kunden-Artikel-Bestellungsbeispiel erweitern/umbauen mit einem Datenimport aus einer csv, json Datei
- ▶ Kunden-Artikel-Bestellungsbeispiel erweitern/umbauen mit einem Datenexport aus einer csv, json Datei

# Ablauf 2026.01.14

- ▶ JSON HÜ ansehen
- ▶ Wiederholungs-Stunde, Hausübungen in Git
- ▶ git Wiederholung

# Ablauf 2026.01.21

- ▶ Test bis 08:45
- ▶ ...danach entlassen zu H.Abart

# Ablauf 2026.01.28

- ▶ Testrückgabe
- ▶ JSON HÜ ansehen
- ▶ eigene GIT-Repo auf Vollständigkeit überprüfen
- ▶ JOINs [https://www.w3schools.com/mysql/mysql\\_join.asp](https://www.w3schools.com/mysql/mysql_join.asp)
- ▶ HÜ Bestellaufgabe so erweitern, an einer Stelle JOINs verwenden

# Ablauf 2026.02.04

- ▶ JOIN HÜ Aufgabe ansehen
- ▶ selfJOIN [https://www.w3schools.com/mysql/mysql\\_join\\_self.asp](https://www.w3schools.com/mysql/mysql_join_self.asp)
- ▶ GROUP BY [https://www.w3schools.com/mysql/mysql\\_groupby.asp](https://www.w3schools.com/mysql/mysql_groupby.asp)
- ▶ GROUP BY [https://www.w3schools.com/mysql/mysql\\_having.asp](https://www.w3schools.com/mysql/mysql_having.asp)
- ▶ UNION [https://www.w3schools.com/mysql/mysql\\_union.asp](https://www.w3schools.com/mysql/mysql_union.asp)
- ▶ UNION ALL [https://www.w3schools.com/mysql/mysql\\_union\\_all.asp](https://www.w3schools.com/mysql/mysql_union_all.asp)
- ▶ Unterschied Union vs JOIN

# Ablauf 2025.02.18

- ▶ WITH: <https://www.tutorialspoint.com/mysql/mysql-common-table-expression.htm>
- ▶ subquery BSP

```
1 SELECT
2   first_name ,
3   last_name ,
4   salary
5 FROM employee e1
6 WHERE salary >
7   (SELECT AVG(salary )
8     FROM employee e2
9     WHERE e1.department_id = e2.department_id )
```

- ▶ Aufgabe Group BY, UNION, WITH, SUBQUERY in ein Beispiel einbauen.

# Ablauf 2025.02.25

- ▶ Tests zurückgeben
- ▶ HÜ ansehen
- ▶ Normalformen <https://ausbildung-in-der-it.de/lexikon/normalisierung-datenbank>
- ▶ [https://www.sn.schule.de/~lava/datenbanken/normalisierung\\_am\\_beispiel.html](https://www.sn.schule.de/~lava/datenbanken/normalisierung_am_beispiel.html)