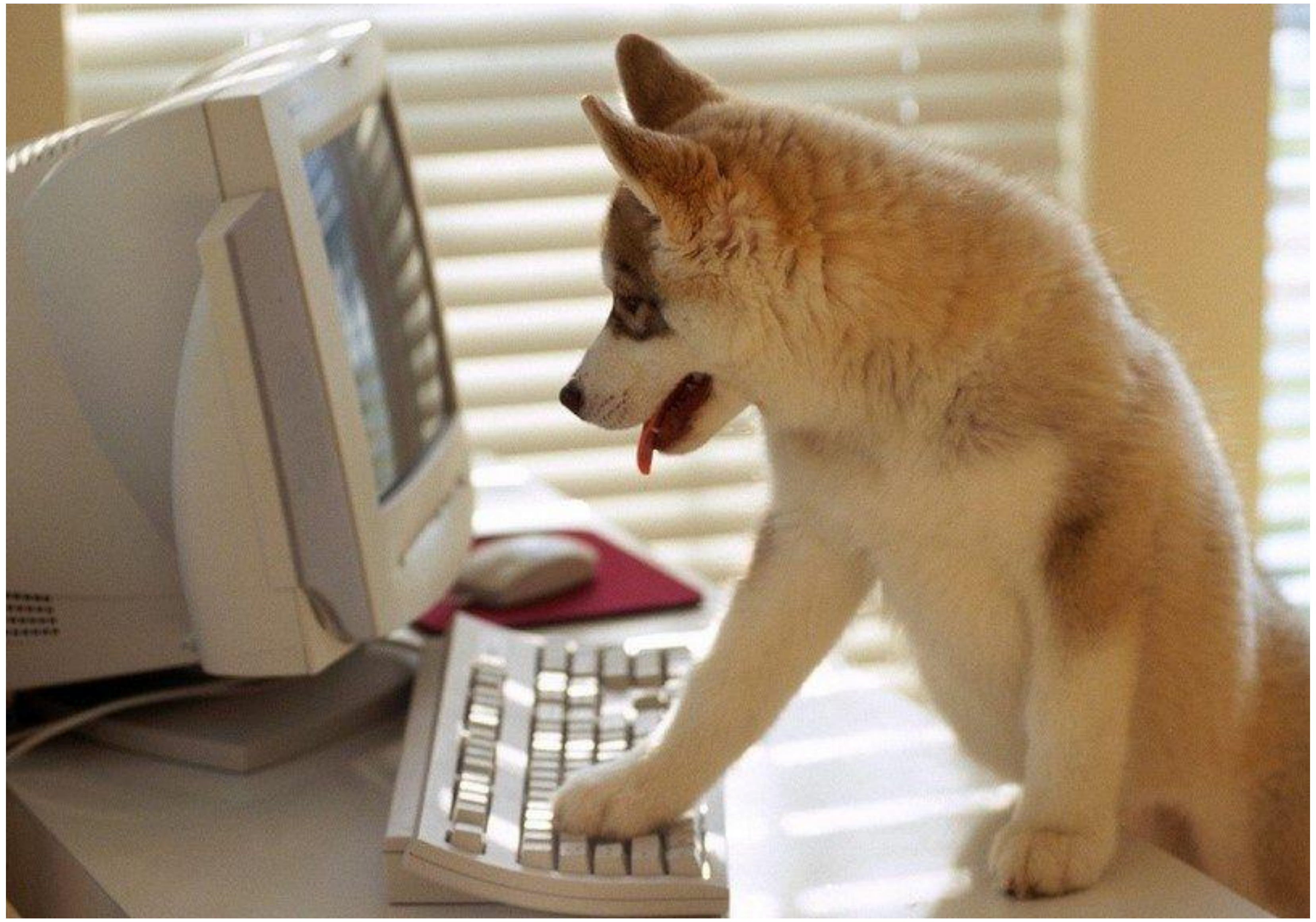
Programming Tutorial [Advanced]



Databases

Database: *“A database is an organized collection of data, generally stored and accessed electronically from a computer system.”* (Wikipedia)

Relational Database: Tables consisting of rows and columns

Structured Query Language (SQL): Domain-specific language for managing relational databases

Databases

Database: *“A database is an organized collection of data, generally stored and accessed electronically from a computer system.”* (Wikipedia)

Relational Database: Tables consisting of rows and columns

Structured Query Language (SQL): Domain-specific language for managing relational databases

In this course we will be using SQLite, which is a cut-down version of SQL

Lambda Expressions

Student		
Matr.No.	**Name**	**Birthday**
12345	John Doe	01.01.95
11111	Jane Doe	02.02.95
10101	Oliver Twist	10.12.93

Lambda Expressions

Professor		
ID	**Name**	**Birthday**
54321	Albert Einstein	14.03.79
12121	Steven Hawking	08.01.42
67890	Bill Gates	28.10.55

Lambda Expressions

Lectures		
ID	**Name**	**Professor**
53214	Relativity Theory	54321
21212	Astrophysics	12121
90876	Computer Science	67890

Lambda Expressions

attendsLecture	
ID	**Matrnr**
54321	12345
12121	11111
67890	10101

Databases

```
CREATE TABLE students(
  MatrNo INTEGER PRIMARY KEY,
  Name TEXT,
  Birthday INTEGER
);
```

Databases

```
INSERT INTO STUDENTS(
  MatrNo,
  Name,
  Birthday
)
VALUES(
  12345,
  “John Doe”,
  010195
);
```

Databases

```
CREATE TABLE lectures(
  ID INTEGER NOT NULL,
  Title TEXT NOT NULL,
  Prof_ID INTEGER NOT NULL,
  PRIMARY KEY(ID),
  FOREIGN KEY(Prof_ID) REFERENCES professors(ID)
  ON DELETE CASCADE ON UPDATE NO ACTION
);
```

Databases

```
INSERT INTO Lectures(
  ID,
  Title,
  Prof_ID
)
VALUES(
  53214,
  “Relativity Theory”,
  54321
);
```

Databases

```
CREATE TABLE attendsLecture(
  MatrNo INTEGER NOT NULL,
  ID  INTEGER NOT NULL,
  PRIMARY KEY(MatrNo,ID),
  FOREIGN KEY(MatrNo) REFERENCES students(MatrNo)
  ON DELETE CASCADE ON UPDATE NO ACTION,
  FOREIGN KEY(Prof_ID) REFERENCES professors(ID)
  ON DELETE CASCADE ON UPDATE NO ACTION
);
```

Databases

```
SELECT *
FROM Students
```

Databases

```
SELECT *
FROM Students
WHERE ID != 11111
```

Databases

```
SELECT *
FROM Students
WHERE ID != 11111
ORDER BY Name DESC
```

Databases

```
SELECT *
FROM Students
WHERE ID != 11111
ORDER BY Name DESC
LIMIT 1
```

Databases

```
SELECT p.name, s.name
FROM Students s, Professors p, attendsLecture a
WHERE s.matrnr = a.matrnr and p.id = a.id;
```

Databases

In this course we will use Github Classroom

1. Get a Github Account if you don’t have one
2. Go to: https://classroom.github.com/a/-LWBrP4R (or scan the QR Code with your phone)
3. Authorize Github and accept the assignment
4. Click on the repository