Databases

The data you need will often live in databases, systems designed for efficiently storing and querying data.

The bulk of these are relational databases, such as Oracle, MySQL, and SQL Server

There systems store data in tables and are typically queried using Structured Query Language (SQL), a declarative language for manipulating data.

Databases

A relational database is a collection of tables (and of relationships among them).

A table is simply a collection of rows and columns, very similar to pandas DataFrames.

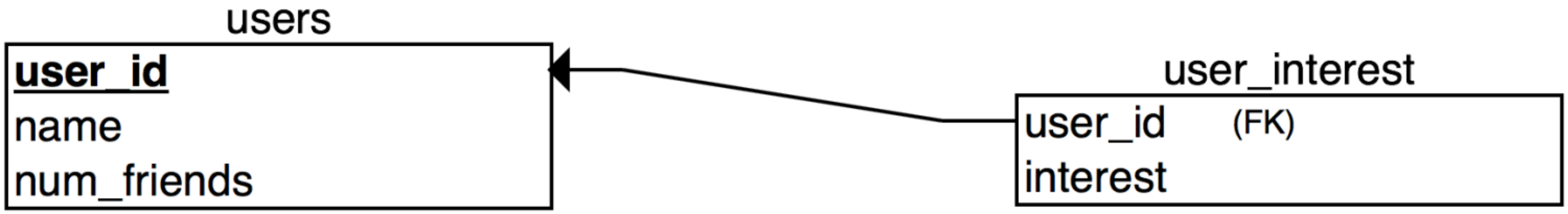
However, each table has to have at least one column called the (primary) key or a foreign key

These special columns relate different tables to each other - therefore the name ‘Relational Database’.

Tables together with key/foreign key relationships are called the schema of a DB

A Simple DB

Consider a simple DB of people and their interests



users

user_id
name
num_friends

user_interest

user_id (FK)
interest

user_id	name	num_friends
0	Hero	0
1	Dunn	2
2	Sue	3
3	Chi	3
4	Thor	3

user_id	interest
0	SQL
0	NoSQL
2	SQL
2	MySQL

SQL

SQL (Structured Query Language) is a domain-specific language used in programming and designed for managing data held in a relational database management system (RDBMS).

SQL was one of the first commercial languages for Edgar F. Codd's relational model, as described in his influential 1970 paper, "A Relational Model of Data for Large Shared Data Banks." Despite not entirely adhering to the relational model as described by Codd, it became the most widely used database language.

SQL

From our perspective, the most important is the ‘SELECT’ statement that allows you to extract data from the DB tables:

```
SELECT * FROM users;                -- get entire contents of table ‘users’
SELECT * FROM users LIMIT 2;        -- get the first two rows
SELECT user_id FROM users;          -- get column ‘user_id’ of table ‘users’
SELECT user_id FROM users WHERE name = 'Dunn';
                                    -- get ‘user_id’ for row ‘Dunn’
```

SQL

You can also use SELECT statements to calculate columns:

```
SELECT LENGTH(name) AS name_length FROM users;
```

PyMySQL

Here is a very simple Python program that accesses a DB:

```
import PyMySQL

# Open database connection
db = PyMySQL.connect("somehost","someuser","somepasswd","USERDB" )

# prepare a cursor object using cursor() method
cursor = db.cursor()

# execute SQL query using execute() method.
cursor.execute("SELECT VERSION()")

# Fetch a single row using fetchone() method.
data = cursor.fetchone()

print ("Database version : %s " % data)

# disconnect from server
db.close()
```

PyMySQL

Let’s try to do some data queries with a Python program:

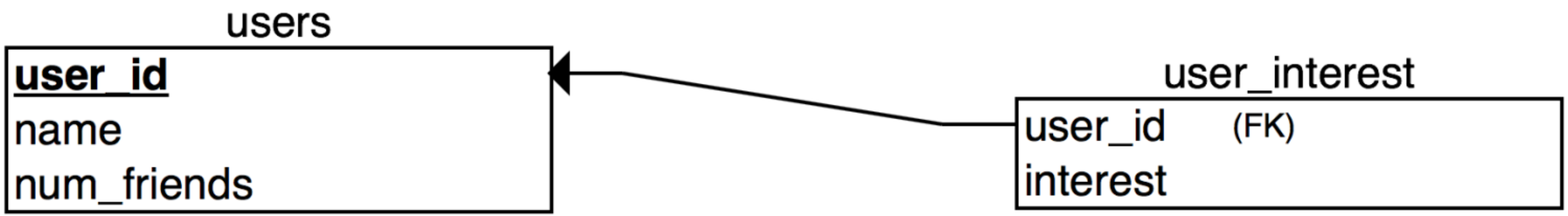
```
import PyMySQL

# Open database connection and prepare a cursor object using cursor() method
db = PyMySQL.connect("somehost","someuser","somepasswd","USERDB" )
cursor = db.cursor()

# SQL query
cursor.execute("SELECT * FROM users”)
results = cursor.fetchall()

# Now print fetched result
for row in results:
    user_id = row[0]
    name = row[1]
    num_friends = row[2]
    print ("user_id= %d,name = %s,num_friends = %d" % (user_id, name, num_friends ))

# disconnect from server
db.close()
```

JOIN

In our schema users and their interests was stored across two different tables which are related via a foreign key.

This is called normalizing a schema in order to avoid redundancy.

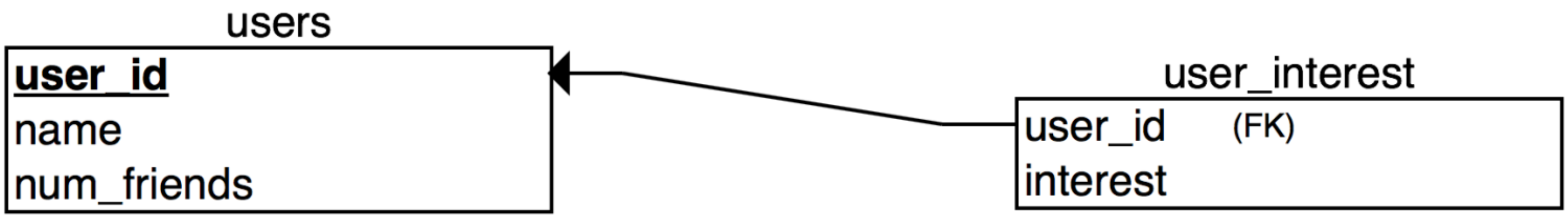
Now, if we wanted to find all users with an interest in SQL we will need to perform a query across two tables - a JOIN



JOIN

Here is the query:

```
SELECT users.name
FROM users
JOIN user_interest
ON users.user_id = user_interest.user_id
WHERE user_interest.interest = 'SQL'
```



JOIN

Let’s try to do this with a Python program:

```
...
# SQL query
sql = ‘’’
SELECT users.name
FROM users
JOIN user_interest
ON users.user_id = user_interest.user_id
WHERE user_interest.interest = 'SQL'
‘’’
cursor.execute(SQL)
results = cursor.fetchall()

# Now print fetched result
for row in results:
    name = row[0]
    print (“name = %s" % name)
...
```

SQL

What we saw is just a tiny sliver of the full SQL.

SQL consists of a data definition language, data manipulation language, and data control language.

The scope of SQL includes data insert, query, update and delete, schema creation and modification, and data access control.

Final

Final due date will be announced via Sakai by the end of this week.

No late submissions will be accepted - no exceptions!

THE END