SQLite3 and Python Primer

CMPSC 305 – Database Systems

ALLEGHENY COLLEGE

Big Data

- Upwards of 2.7 Zetabytes of data exist in the digital universe
- YouTube users upload 48 hours of new video every minute
- Increase in unstructured data: text, photos, etc. https://www.waterfordtechnologies.com/big-data-interesting-facts/

Multiples of bytes V•T•E

Decimal			Binary			
Value	Metric		Value	IEC		JEDEC
1000	kB	kilobyte	1024	KiB	kibibyte	KB kilobyte
1000^2	MB	megabyte	1024^2	MiB	mebibyte	MB megabyte
1000^3	GB	gigabyte	1024^3	GiB	gibibyte	GB gigabyte
1000^4	TB	terabyte	1024^4	TiB	tebibyte	–
1000^5	PB	petabyte	1024^5	PiB	pebibyte	–
1000^6	EB	exabyte	1024^6	EiB	exbibyte	–
1000^7	ZB	**zettabyte**	1024^7	ZiB	zebibyte	–
1000^8	YB	yottabyte	1024^8	YiB	yobibyte	–

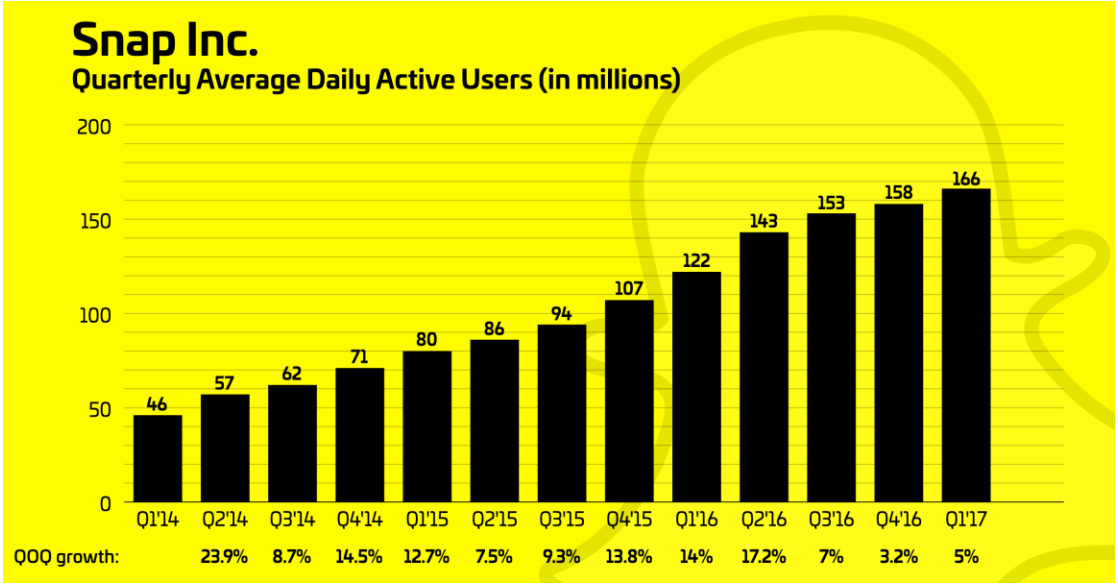
Orders of magnitude of data

Facebook's Daily Data Use

Facebook processes:

- 2.5 billion pieces of content
- Upwards of 500 terabytes of data each day from status and location details
- Processing in 2.7 billion Like actions
- 300 million photos per day
- Scans roughly 105 terabytes of data each half hour
- 100 petabytes of data are stored in a single Hadoop disk cluster (a distributed system for data management)

Current Estimates for Users Online



- Facebook: 2.7 Billion Active users
- Amazon: 112 Million (US users)
- SnapChat: 238 million daily active users worldwide
- Google: 4.39 Billion internet users (worldwide)
- Instagram: 1 Billion monthly active users, 500 Million each day.

Lots of names, photos, passwords and posts to record!

How are we to manage all this data?

Automate the database management processes using software!!

Standardized Database Access with Python

PEP 0249

- Python Database API Specification v2.0
- https://www.python.org/dev/peps/pep-0249/
- A standard API to encourage similarity between the Python modules used for accessing databases.
- Does not provide a library nor a module, just specifications on how to make them
- Third party modules may adhere to these specifications

Steps to run a command in SQL using Python

Five basic steps to using a database according to the Python Database API Specification v2.0

Building automated framework in Python3

- Step 1: Defining the query
- Step 2: Connecting to the database
- Step 3: Execute the query
- Step 4i, (SELECT): Analyze the result
- Step 4ii, or (UPDATE): Commit the change
- Step 5: Cleaning up; close the database connection

Nice tutorial: http://sebastianraschka.com/Articles/2014_sqlite_in_python_tutorial.html


KEEP
CALM
AND
LET'S
CODE

Making Useful Strings

A concatenated string

Note the 'f' before the quotes to enable formatting

```
myCollege_str = "Allegheny"
mesg_str = f"I go to {myCollege_str }!!"
print(mesg_str)
```

```
myCollege_str = "Allegheny"
myMajor_str = "CompSci"
mesg_str = f"At {myCollege_str}, my major is {myMajor_str}"
print(mesg_str)
```

Adding quotes: note the forward slashes in strings

```
iSay_str = "Cool"
mesg_str = f"They say it is a \"{iSay_str}\" major"
print(mesg_str)
```

Making Useful Strings

A concatenated string

- Queries are strings of code that can be created by Python.
- These queries can be sent to database management software

Making a Query Statement

```
a1_str = "deptName"
a2_str = "course"
name_str = "Miller"
table_str = "Instructor"
```

```
myQuery_str = f"SELECT {a1_str}, {a2_str} FROM {table_str} WHERE name == \"{name_str}\""
```

```
print(myQuery_str)
```

Making Useful Strings

A concatenated string

Making An Insert Statement

```
myTable = "Instructor"
PersonID = "10101"
name_str = "Miller"
student = "S1"
```

```
insert_str = f"INSERT INTO {myTable} VALUES({PersonID}, \"{name_str}\", \"{student}\")"
```

```
print(insert_str)
```

Choose variable names that make sense to your code!

Setting Up Virtual Environment

- Create a project directory

```
mkdir week08
cd week08
```

- Create virtual environment using Python

```
python3 -m venv myenv
# see the file tree
find . -not -path '*/\.*'
```

- Activate myenv the virtual environment

```
source myenv/bin/activate  # macOS/Linux
myenv\Scripts\activate     # Windows
```

- Deactivate the virtual environment

```
deactivate
```

Python: A Database Management System (DMS)

Let's Try It Out!

- Locate the sandbox database builder file sandbox/campusDB build.txt and make your DB.
- Call up your favourite editor and let's begin programming.

Now Modify Your DMS!

Do Something Different!

- Try adding query code for other tables.
- What attributes can you query?
- Can you write code for a query involving two tables?