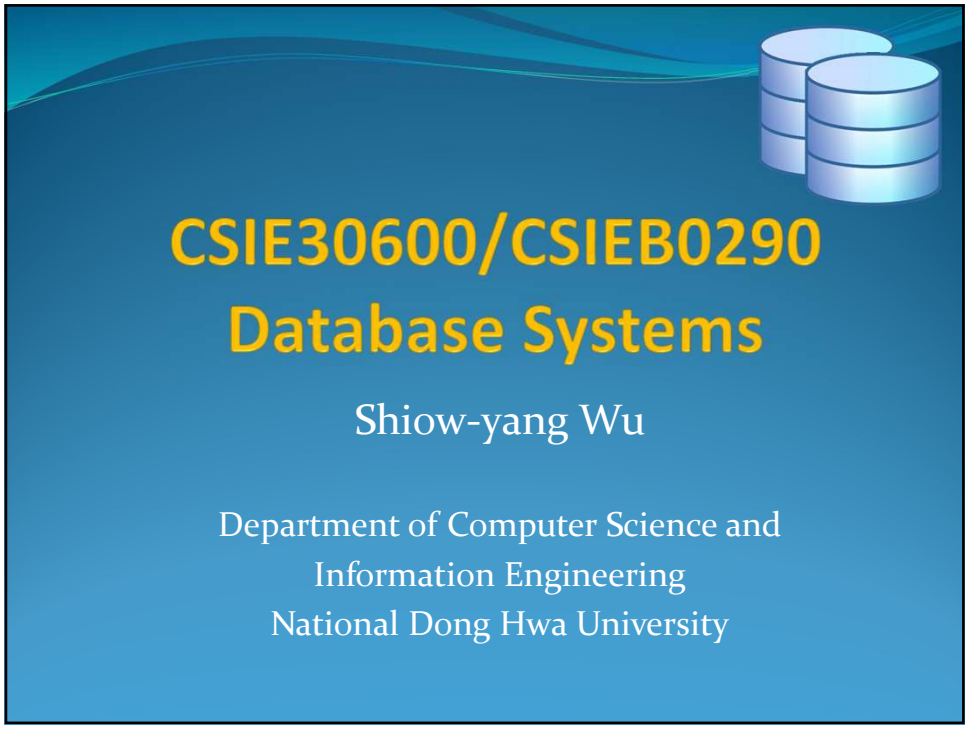


Course Objectives

- First course in database systems
- Cover the fundamental concepts
- Using Database Management System (DBMS).
- Study the internals of DBMS
- Learning state-of-the-art open source DBMS
- **Advanced topics**: Cloud DBs, Big Data, NoSQL, NewSQL, Streaming Data*
- **Advanced topics**: XML DB, OODB, mobile DB, multimedia DB, parallel/distributed DB ...

CSIE30600/CSIEB0290 Database Systems Course Information 2

Course Information

- Instructor's Office: Eng Building C308
- Office Hours: Tue 17:00 - 18:00
- Phone Number: (03) 8693020
- Email Address: showyang@gms.ndhu.edu.tw
- Grading Policy: (may change if necessary)
  - Assignments 25%
  - Midterm 25%
  - Final Exam 25%
  - Term project 25%

Web Pages

- Course web page: http://web.csie.ndhu.edu.tw/showyang/DB2020f/index.html
- Not on "東華e學苑" !!
- Instructor's homepage: http://web.csie.ndhu.edu.tw/showyang

Textbooks

- Avi Silberschatz, Henry F. Korth and S. Sudarshan. ***Database System Concepts, 7th Edition*****.** McGraw-Hill, 2019/2020. (https://www.db-book.com/db7/)
- Ramez Elmasri and Shamkant B. Navathe. ***Fundamentals of Database Systems, 7th Edition***. Pearson, 2016. (https://www.pearson.com/us/higher-education/program/Elmasri-Fundamentals-of-Database-Systems-7th-Edition/PGM189052.html)
- Recommended but not required.

References

- Thomas Connolly and Carolyn Begg. ***Database Systems- A Practical Approach to Design, Implementation, and Management, 6th Edition***. Pearson, 2015.
- Abraham Silberschatz, Henry F. Korth, and S. Sudarshan. ***Database System Concepts, 6th Edition***. McGraw-Hill, 2011.
- Garcia-Molina, J. D. Ullman, and J. Widom. ***Database Systems: The Complete Book, 2nd Edition***, Prentice Hall, 2008.
- Jeffrey D. Ullman and Jennifer Widom. ***A First Course in Database Systems, 3rd Edition***, Prentice Hall, 2007.

SQL References

- Upom Malik, Matt Goldwasser, Benjamin Johnston. ***SQL for Data Analytics: Perform fast and efficient data analysis with the power of SQL***. Packt Publishing, 2019.
- Andrew Johansen. ***SQL: The Ultimate Beginner's Guide!*** CreateSpace Independent Publishing Platform, Nov 2015.
- John Viescas and Michael J. Hernandez. ***SQL Queries for Mere Mortals: A Hands-On Guide to Data Manipulation in SQL (3rd Edition).*** Addison-Wesley Professional, Jun 2014.
- Joe Celko. ***Joe Celko's SQL for Smarties, 5th Edition: Advanced SQL Programming***. Morgan Kaufmann. Dec 2014.
- Stephane Faroult. ***SQL Success – Database Programming Proficiency***. RoughSea Ltd, 2013.
- Ben Forta. ***SQL in 10 Minutes, Sams Teach Yourself, 4th Edition***. Sam Publishing, Nov 2012.

SQL References (cont.)

- Mike McGrath. ***SQL in Easy Steps, 3rd Edition***. In Easy Steps Ltd. 2012.
- Alan Beaulieu. ***Learning SQL, 2nd Edition.*** O'Reilly Media, Inc. 2009.
- James R. Groff, Paul N. Weinberg, Paul Weinberg, James Groff. ***SQL: The Complete Reference, 3rd Edition***. McGraw-Hill, 2009.
- Kevin Kline, Daniel Kline and Brand Hunt. ***SQL In A Nutshell, 3rd Edition.*** O'Reilly Media, Inc. 2008.
- Alex Kriegel and Boris M. Trukhnov. ***SQL Bible, 2nd Edition.*** Wiley, 2008.

PHP and MySQL References

- Robin Nixon. ***Learning PHP, MySQL & JavaScript, 5th Edition***. O'Reilly Media, Inc., 2018.
- Luke Welling and Laura Thomson. ***PHP and MySQL Web Development, 5th Edition***, Addison-Wesley Professional, 2016.
- W. J. Gilmore. ***Beginning PHP and MySQL: From Novice to Professional, 5th Edition***, Apress, 2016.
- Mr Andrew Comeau and Stephen Burge. ***MySQL Explained: Your Step By Step Guide***, CreateSpace Independent Publishing Platform, Nov 2015.
- Paul DuBois. ***MySQL, 5th Edition (Developer's Library).*** Addison-Wesley Professional, 2013.
- Alan Forbes. ***The Joy of PHP: A Beginner's Guide to Programming Interactive Web Applications with PHP and mySQL, 2nd Edition***, Plum Island Publishing LLC, 2013.

Python Programming Books

- Eric Matthes. *Python Crash Course, 2nd Edition: A Hands-On, Project-Based Introduction to Programming*, No Starch Press, 2019.
- Steve Holden, Anna Ravenscroft and Alex Martelli. *Python in a Nutshell, 3rd Edition*. O'Reilly Media, Inc. 2017.
- Wes McKinney. *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and Ipython, 2nd Edition*. O'Reilly Media, 2017.
- Luciano Ramalho. *Fluent Python*. O'Reilly Media, 2015.
- Mark Lutz. *Learning Python, 5th Edition*. O'Reilly Media, 2013.
- David Beazley and Brian K. Jones. *Python Cookbook, 3rd edition*. O'Reilly Media, 2013.

On-line References

- Wikibooks, **Structured Query Language**. (https://en.wikibooks.org/wiki/Structured_Query_Language) (SQL:2011)
- Wikibooks, **MySQL**. (https://en.wikibooks.org/wiki/MySQL)
- Wikibooks, **PostgreSQL** (https://en.wikibooks.org/wiki/PostgreSQL)

On-line SQL Resources

- **Online SQL interpreter (for the DB Concepts book)** (https://www.db-book.com/db7/university-lab-dir/sqljs.html)
- **SQLite online** (https://sqliteonline.com/)
- **The Try-SQL Editor (w3schools)** (https://www.w3schools.com/sql/trysql.asp?filename=trysql_op_in)
- **Execute SQL Online (tutorialspoint)** (https://www.tutorialspoint.com/execute_sql_online.php)

Individual Term Project

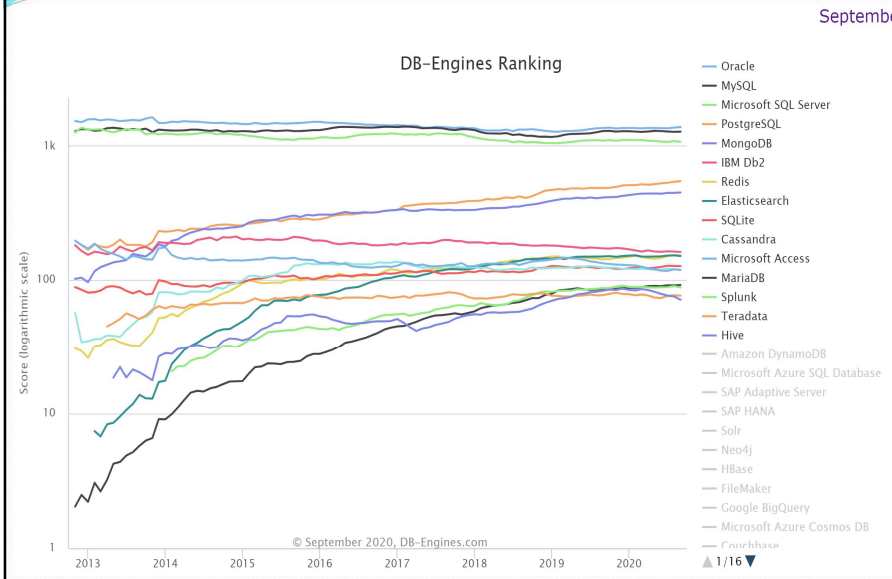
- An on-line database application (details will be announced in class)
- Use an open source DB as backend database
- Use browser or smart phone as user interface
- Can use any technique to connect the database.
- We will discuss PHP+MySQL and/or Python+PostgreSQL.
- Demonstration and report due date: Jan 8, 2021.

Why MySQL, PostgreSQL, SQLite?

358 systems in ranking, September 2020

Rank Sep 2020	Rank Aug 2020	Rank Sep 2019	DBMS	Database Model	Score Sep 2020	Score Aug 2020	Score Sep 2019
1.	1.	1.	Oracle	Relational, Multi-model	1369.36	+14.21	+22.71
2.	2.	2.	MySQL	Relational, Multi-model	1264.25	+2.67	-14.83
3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	1062.76	-13.12	-22.30
4.	4.	4.	PostgreSQL	Relational, Multi-model	542.29	+5.52	+60.04
5.	5.	5.	MongoDB	Document, Multi-model	446.48	+2.92	+36.42
6.	6.	6.	IBM Db2	Relational, Multi-model	161.24	-1.21	-10.32
7.	7.	↑ 8.	Redis	Key-value, Multi-model	151.86	-1.02	+9.95
8.	8.	↓ 7.	Elasticsearch	Search engine, Multi-model	150.50	-1.82	+1.23
9.	9.	↑ 11.	SQLite	Relational	126.68	-0.14	+3.31
10.	↑ 11.	10.	Cassandra	Wide column	119.18	-0.66	-4.22
11.	↓ 10.	↓ 9.	Microsoft Access	Relational	118.45	-1.41	-14.26
12.	12.	↑ 13.	MariaDB	Relational, Multi-model	91.61	+0.69	+5.54
13.	13.	↓ 12.	Splunk	Search engine	87.90	-2.01	+0.89
14.	14.	↑ 15.	Teradata	Relational, Multi-model	76.39	-0.39	-0.57
15.	15.	↓ 14.	Hive	Relational	71.17	-4.12	-11.93
16.	16.	↑ 18.	Amazon DynamoDB	Multi-model	66.18	+1.43	+8.36
17.	17.	↑ 25.	Microsoft Azure SQL Database	Relational, Multi-model	60.45	+3.60	+32.91
18.	18.	↑ 19.	SAP Adaptive Server	Relational	54.01	+0.05	-2.09
19.	19.	↑ 21.	SAP HANA	Relational, Multi-model	52.86	-0.26	-2.53
20.	20.	↓ 16.	Solr	Search engine	51.62	-0.08	-7.35

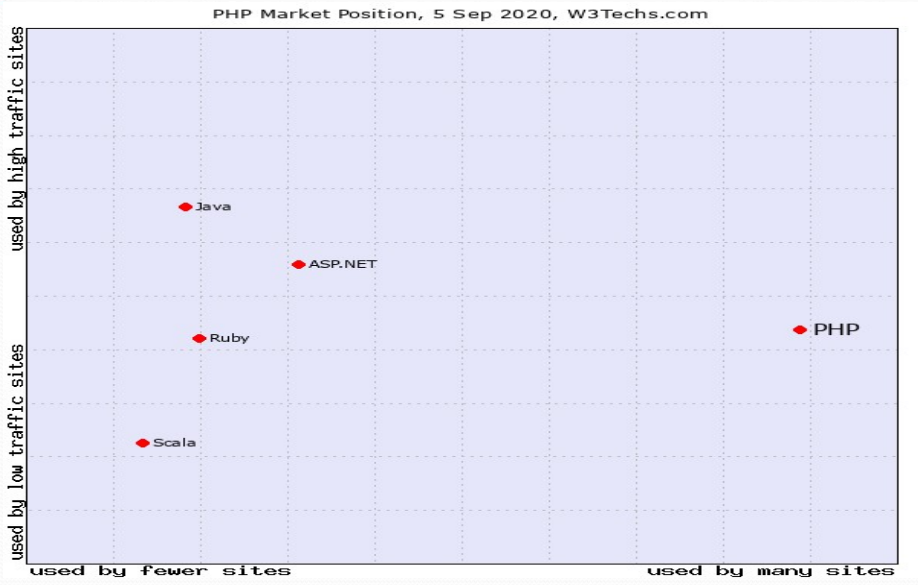
Why MySQL, PostgreSQL, SQLite?



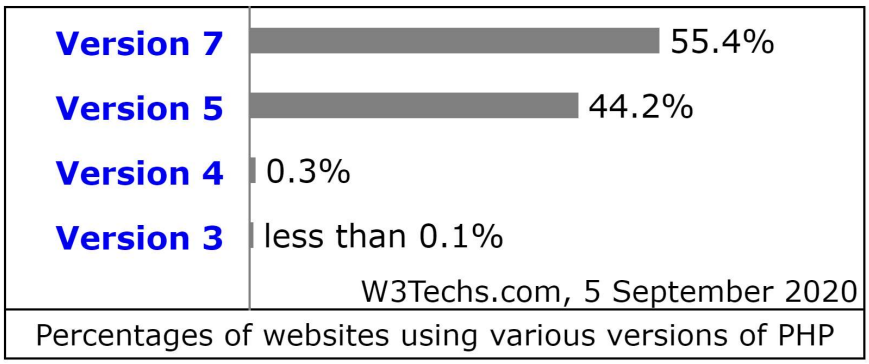
Why PHP ? (Worldwide Usage)

- Percentage of websites using PHP (**78.9**%)

Why PHP ? (Market Position)



Which Version of PHP?



Why Python ? (TIOBE Index)

Aug 2020	Aug 2019	Change	Programming Language	Ratings	Change
1	2	↑	C	16.98%	+1.83%
2	1	↓	Java	14.43%	-1.60%
3	3		Python	9.69%	-0.33%
4	4		C++	6.84%	+0.78%
5	5		C#	4.68%	+0.83%
6	6		Visual Basic	4.66%	+0.97%
7	7		JavaScript	2.87%	+0.62%
8	20	⇈	R	2.79%	+1.97%
9	8	↓	PHP	2.24%	+0.17%
10	10		SQL	1.46%	-0.17%
11	17	⇈	Go	1.43%	+0.45%
12	18	⇈	Swift	1.42%	+0.53%
13	19	⇈	Perl	1.11%	+0.25%
14	15	↑	Assembly language	1.04%	-0.07%
15	11	⇊	Ruby	1.03%	-0.28%
16	12	⇊	MATLAB	0.86%	-0.41%
17	16	↓	Classic Visual Basic	0.82%	-0.20%
18	13	⇊	Groovy	0.77%	-0.46%
19	9	⇊	Objective-C	0.76%	-0.93%
20	28	⇈	Rust	0.74%	+0.29%

Why Python ? (PYPL Index)

Worldwide, Sept 2020 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Python	31.56 %	+2.9 %
2		Java	16.4 %	-3.1 %
3		Javascript	8.38 %	+0.3 %
4		C#	6.5 %	-0.8 %
5		PHP	5.85 %	-0.5 %
6		C/C++	5.8 %	+0.0 %
7		R	4.08 %	+0.3 %
8		Objective-C	2.79 %	+0.2 %
9		Swift	2.35 %	-0.1 %
10		TypeScript	1.92 %	+0.1 %

Why Study Databases?

- Databases used to be *specialized applications*, now they are a *central component* in computing environments
  - Knowledge of database concepts is essential for computer scientists
- Databases are everywhere, even when you don't see them
  - most activities involve data
    - Banking + credit cards: all transactions
    - Airlines: reservations, schedules
    - Universities: registration, grades
    - Telecommunications/networks

(more on next slide)

Why Study Databases?

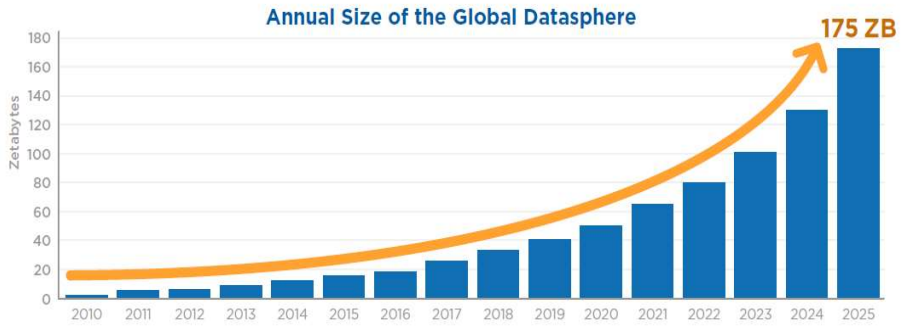
- Sales: customers, products, purchases
- Manufacturing: production, inventory, orders, supply chain
- Human resources: employee records, salaries, tax deductions
- Web sites: generated from databases; front-ends to databases
- Scientific research, e.g., studying the environment
- Your own data!

- Global data volume grows faster than ever! (next slide)
- Sky-high demand for Big data and NoSQL DB!
- Streaming data from IoT devices
- Data needs to be *managed*

Global DataSphere

- DataSphere: the sum of ALL data around the world!
- Global DataSphere to Hit **59** Zettabytes(**ZB**) in 2020 Alone(IDC)
- Global DataSphere will grow to **175 ZB** by 2025 (IDC)

Figure 1 – Annual Size of the Global Datasphere



Why Study Databases?

- Because data is valuable:
  - E.g., bank account records, tax records, student records, personal information …
  - It must be protected - no matter what happens whether we have machine crashes, disk crashes, hurricanes/floods;
  - It also needs to be protected from **people**

Why Study Databases?

- Because data is often structured:
  - Bank account records all follow the same structure
  - We can exploit this regular structure
    - To retrieve data in useful ways (that is, we can use a *query language)*
    - To store data efficiently
- Dealing with unstructured data still needs database technologies.
- Big data needs database + new techniques

Why Study Databases?

- Because the database field has made significant contributions to basic computer science:
  - *Understand concepts and apply to different problems and different areas*
- Because DBMS software is highly successful as a commercial technology (Oracle, DB2, SQL Server...)
- Because DB research is highly active and VERY interesting!
  - Lots of opportunities to have practical impact

Syllabus

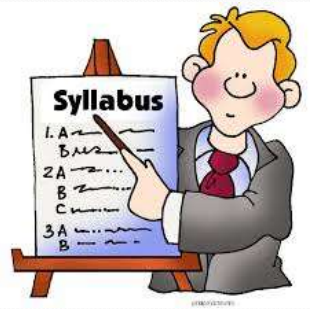
- Introduction
- Databases and database users
- Database system concepts and architecture
- Relational model and constraints
- Relational algebra (and calculus**)
- Basic & intermediate SQL (how to use a DB)
- Database design with ER/EER models
- ER/EER to relational mapping
- Open source RDBMS(MySQL, PostgreSQL, ...)
- Web DB applications(with PHP, Python, ...)

Syllabus (cont.)

- Relational database design I – Functional dependencies and normalization
- Relational database design II – Further normalization and design algorithms
- Big data processing
  - Basic concepts, big data storage, MapReduce
  - NoSQL/NewSQL, graph databases
- Big data analytics
  - Data warehousing
  - Online/realtime analytical processing
  - Data mining

Syllabus (cont.)

- Complex data types**
- Storage systems and structure**
- Indexing methods**
- Query processing & optimization
  - Query processing
  - Query optimization
- Transaction management
  - Transactions
  - Concurrency control
  - Recovery**



Syllabus (cont.)

- Object and object-relational databases**
- Semi-structured data and XML**
- Web databases

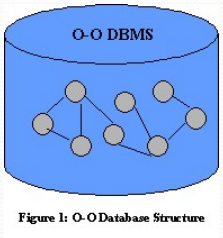


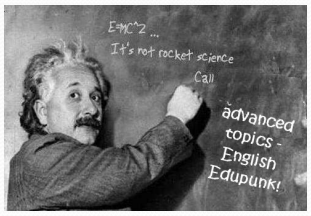
Figure 1: O-O Database Structure

Syllabus (cont.)*

- Parallel and distributed databases
- Cloud computing and data trends
- NoSQL and NewSQL databases
- Blockchain databases

Syllabus (Advanced Topics)*

- Database security
- Active databases
- Temporal and real-time databases
- Spatial databases
- Multimedia databases
- Deductive databases
- Information retrieval and Web search
- Mobile and pervasive data management
- Streaming data management/analytics



Accept the DB Challenges

- A very interesting and challenging class
- Be prepared for some theoretical discussion on the principles and algorithms.
- You must keep up with the pace.

- Ask questions if you miss the point.
- Design your own examples.

- Join us with the DB challenges !!!