

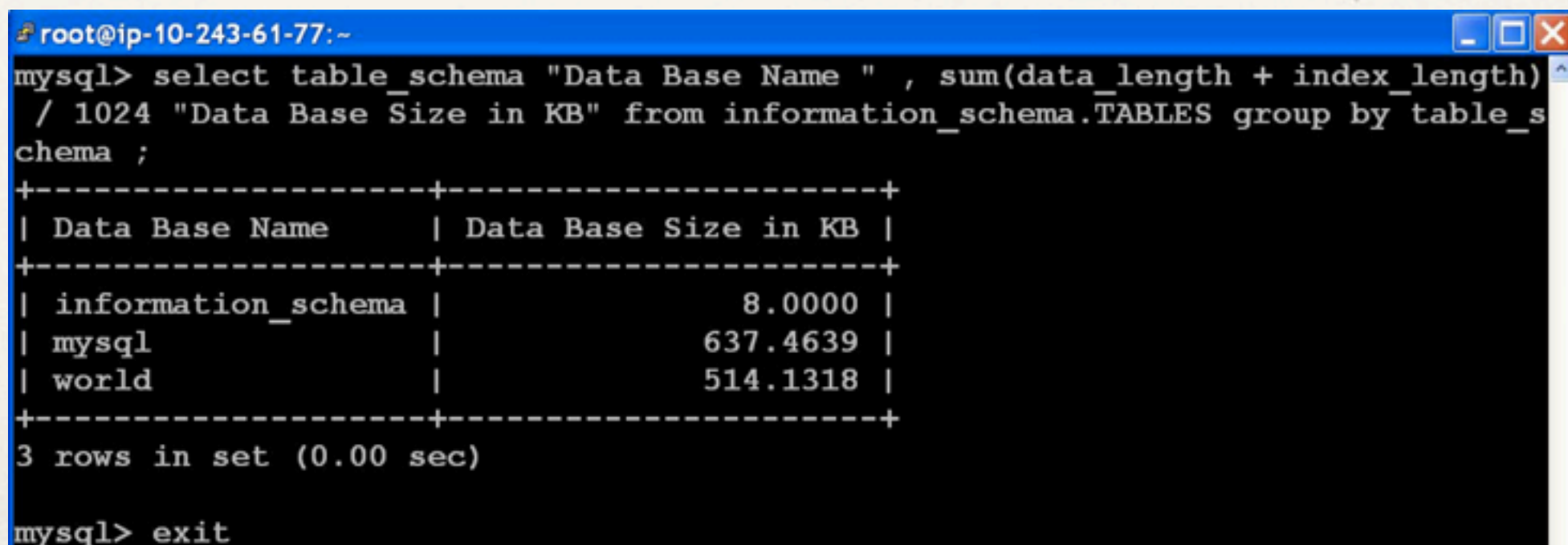
DataBases and SQL

CDIPS Data Science Workshop, Day 4

Jackie Brosamer

SQL

- ❖ Structure Query Language
- ❖ Scripting language to interact efficiently and easily with databases
 - ❖ Can query information from database as well as modify
- ❖ Not used much in academia but VERY widely used in industry



```
root@ip-10-243-61-77:~
mysql> select table_schema "Data Base Name " , sum(data_length + index_length)
/ 1024 "Data Base Size in KB" from information_schema.TABLES group by table_s
chema ;
+-----+-----+
| Data Base Name | Data Base Size in KB |
+-----+-----+
| information_schema | 8.0000 |
| mysql | 637.4639 |
| world | 514.1318 |
+-----+-----+
3 rows in set (0.00 sec)

mysql> exit
```


SQL Tables

games		city	
yr	city	name	country
1896	Athens	Sydney	Australia
1948	London	Athens	Greece
2004	Athens	Beijing	China
2008	Beijing	London	UK
2012	London		

- ❖ Database *Olympics* is a collection of **tables** (*games*, *city*)
- ❖ Table *games* has two **rows**: *yr* (int) and *city* (varchar)

SQL Queries

- ❖ **SELECT**: Which cities have hosted the Olympics?

- ❖ SELECT city FROM games

- ❖ **WHERE**: What years did London host?

- ❖ SELECT yr FROM games WHERE city='London'

- ❖ **JOIN**: Which countries have hosted the Olympics?

- ❖ SELECT country FROM city JOIN games ON city=name

- ❖ **SELECT...WHERE**: Which countries have hosted the Olympics since 2000?

- ❖ SELECT country FROM city JOIN games ON city=name WHERE yr >2000

- ❖ *Bonus: why might joining on a varchar be a bad idea? What would be a better idea from a database design perspective?*

games		city	
yr	city	name	country
1896	Athens	Sydney	Australia
1948	London	Athens	Greece
2004	Athens	Beijing	China
2008	Beijing	London	UK
2012	London		

Tutorials from SQLZoo

1. SELECT basics: http://sqlzoo.net/wiki/SELECT_basics
2. SELECT with aggregate functions: [http://sqlzoo.net/wiki/SUM and COUNT](http://sqlzoo.net/wiki/SUM_and_COUNT)
3. SELECT with JOIN: [http://sqlzoo.net/wiki/More JOIN operations](http://sqlzoo.net/wiki/More_JOIN_operations)
 - Do 1-6 on your own!