>>

COMP9315 Exercises

- COMP9315 21T1
- Doing Prac Work
- Exercise: Unix File I/O (revision)
- Exercise: PostgreSQL files
- Exercise: Table Statistics
- Exercise: Extracting a Schema
- Exercise: Enumerated Types

∧ >>

❖ COMP9315 21T1

Aim: explore how DBMSs work internally

- examine overall architecture
- algorithms and data structures for RA ops
- analyse costs (disk reads/writes)

with examples from the PostgreSQL database.

Why PostgreSQL?

- open-source
- well-written code base

Today, we examine the catalog (via SQL) to see PG meta-data

<< ^ >>

❖ Doing Prac Work

Rule #1: RTFEM ... Read the Error Messages

Rule #2: RTFLF ... Read the Log File

Rule #3: TWEMS ... Try What the Error Message Says

Do the above before posting on the forum.

If *still* can't fix it, email cs9315@cse

- arrange a Teams/Meet/Zoom session *with* screen sharing

<< ^ >>

❖ Exercise: Unix File I/O (revision)

Write a C program that reads a file, block-by-block.

Command-line parameters:

- block size in bytes
- name of input file

Use low-level C operations: **`open, read`**.

Count and display how many blocks/bytes read.

<< ∧ >>

❖ Exercise: PostgreSQL files

Explore the PostgreSQL **`pg_catalog`** to determine ...

How many tables are in the catalog?

What users? namespaces? tablespaces? are there

What information is stored about tables, attributes, types?

How many tables are in the **`public`** schema?

Which directory contains the database *X*?

Which files in that directory correspond to table *Y*?

<< ∧ >>

❖ Exercise: Table Statistics

Using the PostgreSQL catalog, write a PLpgSQL function

- to return table name and #tuples in table
- for all tables in the **`public`** schema

```
create type TableInfo as
    (tabname text, ntuples int);

create function pop()
    returns setof TableInfo
as $$
...
$$ language plpgsql;
```

Hint: you will need to use dynamically-generated queries.

<< ∧ >>

❖ Exercise: Extracting a Schema

Write a PLpgSQL function:

- **`function schema() returns setof text`**
- giving a list of table schemas in the **`public`** schema

It should behave as follows:

```
db=# select * from schema();
            tables
---------------------------
 table1(x, y, z)
 table2(a, b)
 table3(id, name, address)
...
```

<< ∧

❖ Exercise: Enumerated Types

PostgreSQL allows you to define enumerated types, e.g.

```
create type Mood as enum ('sad', 'happy');
```

Creates a type with two ordered values **'sad' < 'happy'**

What is created in the catalog for the above definition?

Hint:

```
pg_type(oid, typname, typelen, typetype, ...)
pg_enum(oid, enumtypid, enumlabel)
```

Produced: 16 Feb 2021