

pg_classy

https://github.com/PgClassy/pg_classy

Typing Is Hard

Typing Is Hard

```
CREATE TABLE customer_status(  
    customer_status_id serial PRIMARY KEY  
    , customer_status text NOT NULL  
);
```

```
CREATE INDEX  
customer_status__u_lower_customer_status ON  
customer_status(lower(customer_status));
```

Typing Is Hard

```
CREATE FUNCTION customer_status__get_text(  
    customer_status_id int  
) RETURNS text LANGUAGE plpgsql STABLE $$
```

```
    blah blah blah
```

```
$$;
```

```
CREATE FUNCTION customer_status__get_id(  
    customer_status text  
) RETURNS int LANGUAGE plpgsql STABLE ....
```

Screw That

Screw That

```
SELECT code.lookup_table( 'customer_status' );
```

Magic!

Wingardium Codeiosa

- `pg_classy` makes use of trunklet; a template system for Postgres
- Define a creation template that accepts user-supplied parameters
- Define a unique identifier template that uniquely identifies an instance of a class by a database object
- Optionally define a pre-processing template

Preprocessing

```
CREATE VIEW customer_v AS  
  SELECT c.*, cs.customer_status  
  FROM customer LEFT JOIN customer_status
```

... so much typing ...

Preprocessing

```
SELECT code.lookup_view('customer');
```

So little typing!

Upgrades!

- You discover citext and what to use it for lookup tables.
- Create an upgraded version of the lookup class.
- New version uses new creation template.
- New version also has an *upgrade template* that will upgrade any existing instances.

WARNING:

This extension contains
alpha content. User
discretion is advised.

[https://github.com/
PgClassy/pg_classy](https://github.com/PgClassy/pg_classy)

Jim.Nasby@OpenSCG.com