ORACLE

Fixing a Schema with Domains and Annotations

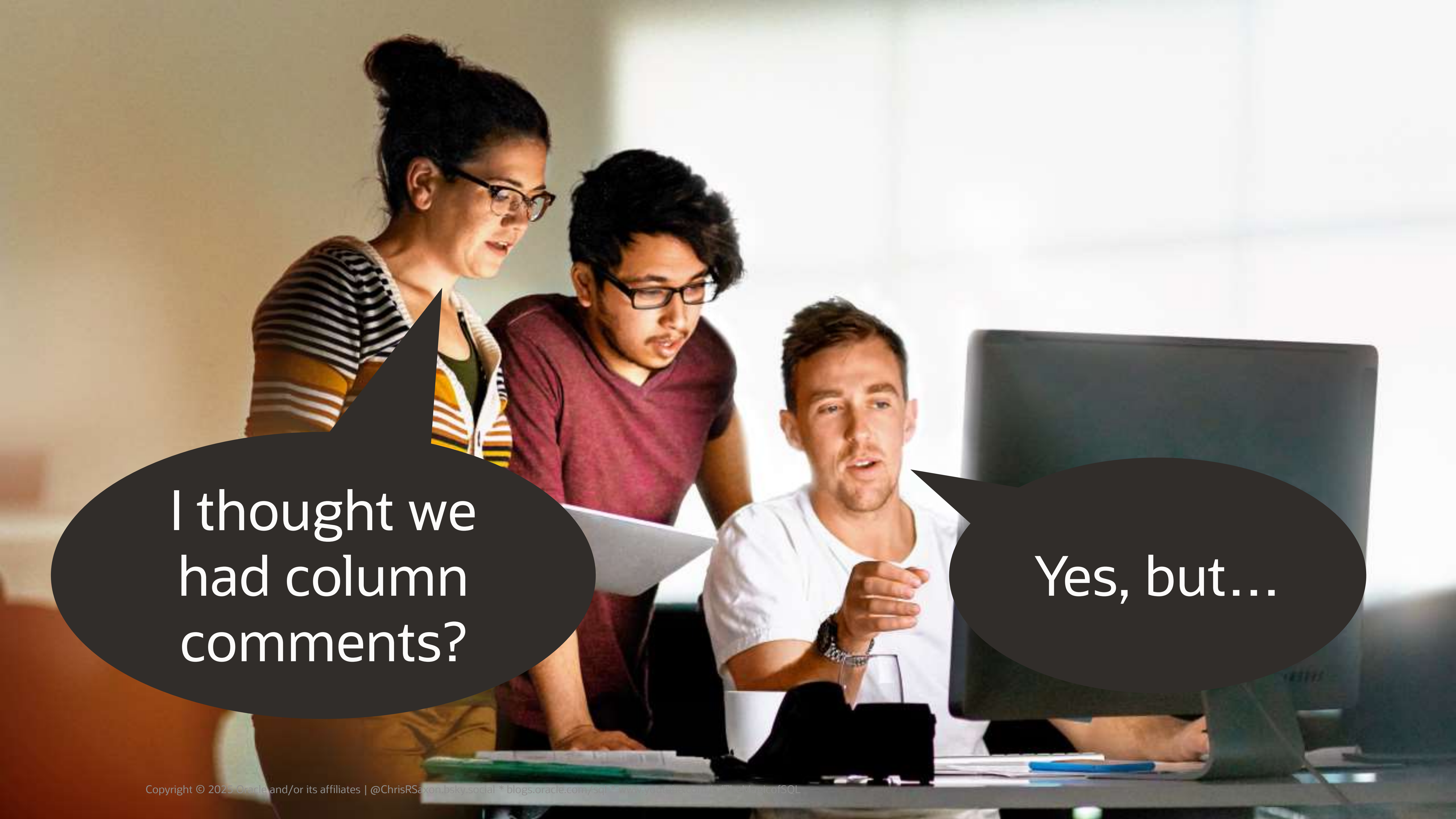
Add data intent to describe messy schemas

Alex Nuijten
@alexnuijten
https://nuijten.blogspot.com/

Chris Saxon, Oracle Developer Advocate
@ChrisRSaxon.bsky.social & @SQLDaily
https://blogs.oracle.com/sql
https://www.youtube.com/c/TheMagicofSQL
https://www.linkedin.com/in/saxonchris/

Say what you want an app to do

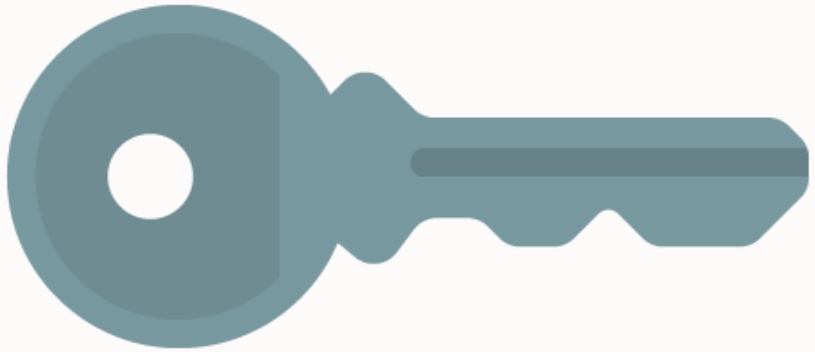
AI generates the app



```
comment on column products.id is 'obvious';
comment on column products.name is
'this means it is the name, I guess; allowed operations:
insert; update; delete; read; display;';
comment on column products.i_date is
'{"meaning":"I don''t know",
  "dml":"delete; read; display;"}';
comment on column products.description is
'<meaning>something about products</meaning>
<operations>insert; update; delete; read; display;
</operations>';
comment on column products.sku is '...';
comment on column products.contacteaddr is '¯\_(ツ)_/¯';
```

Schema Annotations

Metadata for applications



[value]

Annotations are a series of
keys with *optional* values

```
comment on column products.i_date is
'{"meaning":"I don''t know",
  "dml":"delete; read; display;"}';

alter table products modify ( i_date
  annotations (
    system_generated, --default add
    add display 'Insert timestamp',
    replace allowed_select '["filter", "sort"]',
    add or replace disallowed_dml '["insert", "update"]'
  )
)
```

DEMO

Data Intention Language (DIL) - Schema Annotations

Metadata

Key-value pairs

Freeform

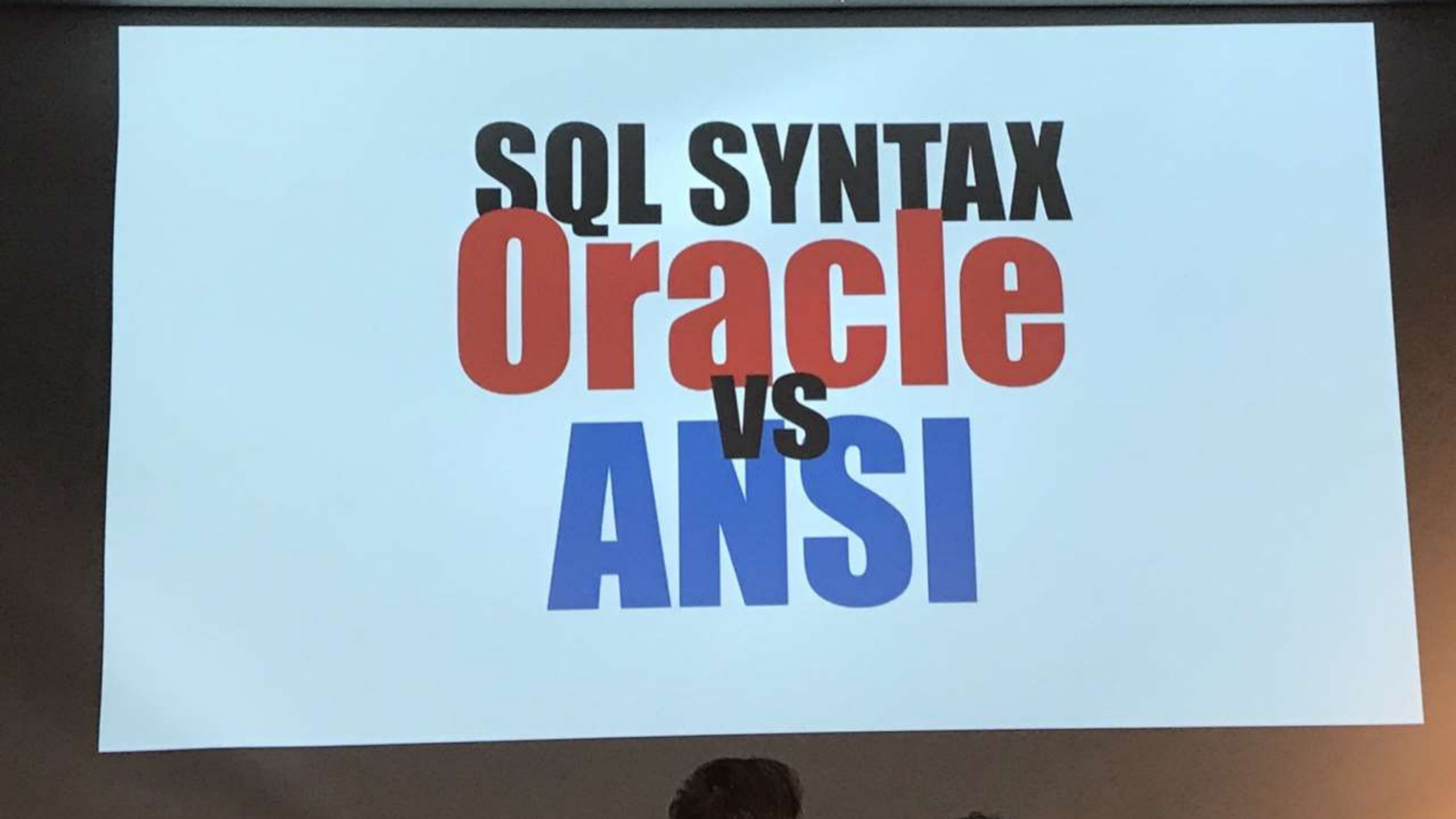
No DB validation

Data Use Case Domains

Extend base types with use case information

```
create [ usecase ] domain <domain_name> as <datatype>
  [ strict ]
  [ <default> ]
  [ <collation> ]
  [ not null ]
  [ <check constraints> ]
  [ <display expr> ]
  [ <order expr> ]
  [ <annotations> ]
```

Optional properties


SQL SYNTAX
Oracle
vs
ANSI


1 - 1

```
create                  domain <domain_name> as <datatype>

   [ <default> ]
   [ <collation> ]
   [ not null ]
   [ <check constraints> ]
```

SQL standard

```
create [ usecase ] domain <domain_name> as <datatype>
  [ strict ]
  [ <default> ]
  [ <collation> ]
  [ not null ]
  [ <check constraints> ]
  [ <display expr> ]
  [ <order expr> ]
  [ <annotations> ]
```

SQL standard

Oracle extension

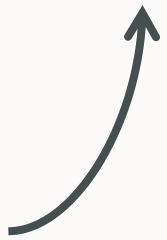
```
create usecase domain insert_timestamp as timestamp


  annotations (
    system_generated,
    display 'Insert timestamp',
    allowed_reads '["filter", "sort"]' ,
    disallowed_writes '["insert", "update"]'
  );
```


```
create domain insert_timestamp as timestamp
  default systimestamp
  not null
  annotations (
    system_generated,
    display 'Insert timestamp',
    allowed_reads '["filter", "sort"]' ,
    disallowed_writes '["insert", "update"]'
  );
```

```
comment on column products.contacteaddr is '¯\_(ツ)_/¯';
```

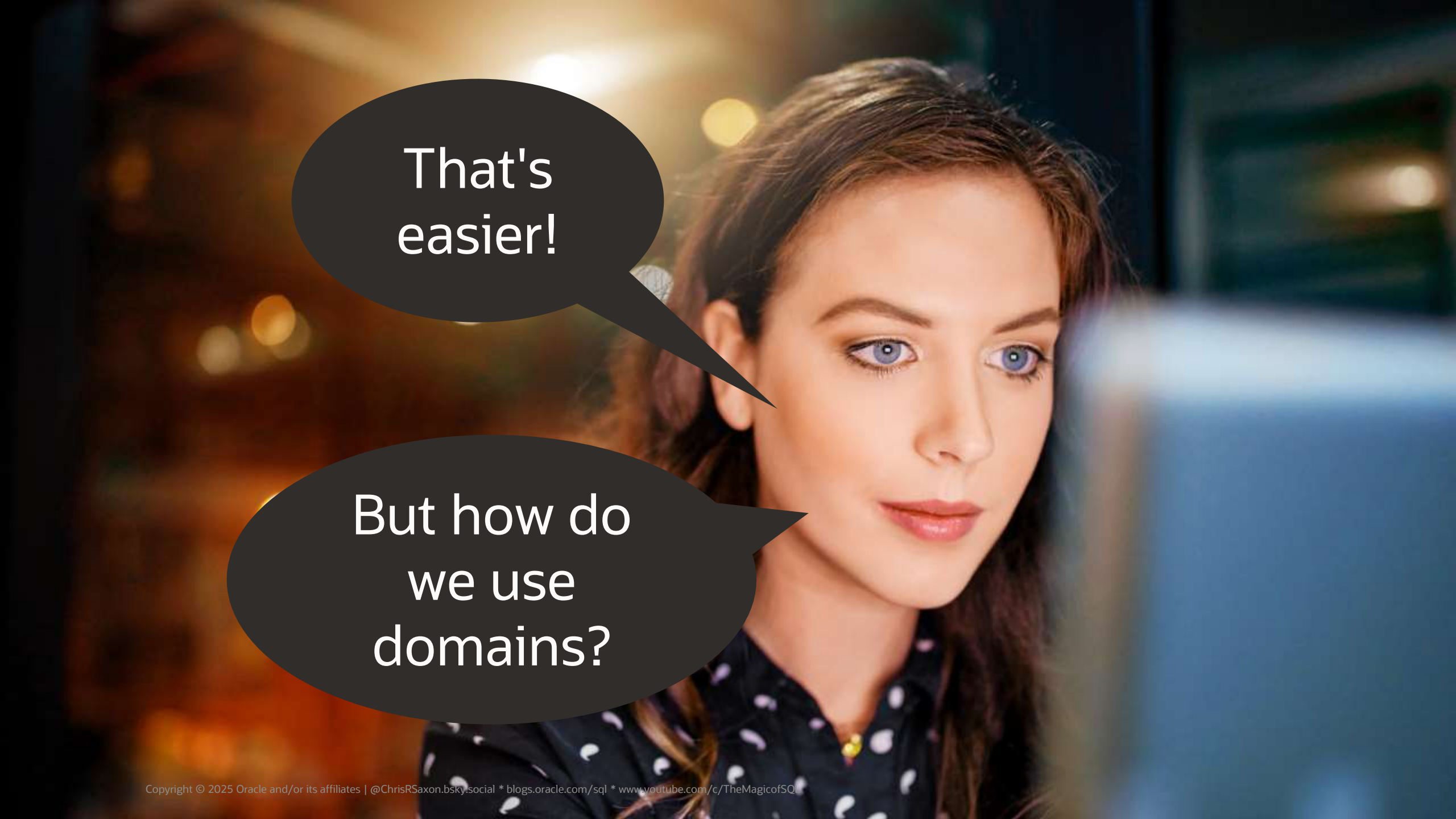
Contact email address




```
REGEXP_LIKE (
   email,
   '^([a-zA-Z0-9!#$%&*+=?^_`{|}~-]+
   (\.[A-Za-z0-9!#$%&*+=?^_`{|}~-]+
   )*)@((a-zA-Z0-9]([a-zA-Z0-9-]*
   [a-zA-Z0-9])?\.)+[a-zA-Z0-9](
   [a-zA-Z0-9-]*[a-zA-Z0-9])?)$'
)
```

```
select domain_name, search_condition
from   dba_domain_constraints
where  domain_owner = 'SYS';

DOMAIN_NAME          SEARCH_CONDITION
-------------------- --------------------------------
PHONE_NUMBER_D       REGEXP_LIKE (phone_number_d,'^…$')
EMAIL_D              REGEXP_LIKE (email_d, '^…$')
DAY_SHORT_D          UPPER(day_short_d) IN (…)
DAY_D                UPPER(day_d) IN (…)
…
```



```
create table customers ( …,
  insert_datetime insert_timestamp
);

create table invoices ( …,
  insert_datetime domain insert_timestamp
);

create table orders ( …,
  insert_datetime timestamp domain insert_timestamp
);
```

SQL standard

domain => Oracle extension

```
alter table products
  modify ( contacteaddr )
  add domain sys.email_d;

alter table orders
  modify ( e_addr domain sys.email_d );
```

Applies constraint, default, etc. to the columns

```
select table_name, search_condition
from   user_constraints
where  domain_name = 'EMAIL_D';

TABLE_NAME     SEARCH_CONDITION
-------------- --------------------------------
ORDERS         REGEXP_LIKE ("E_ADDR",'^...$')
INVOICES       REGEXP_LIKE ("CUSTOMER_EMAIL",'^...$')
CUSTOMERS      REGEXP_LIKE ("EMAIL",'^...$')
PRODUCTS       REGEXP_LIKE ("CONTACTEADDR",'^...$')
```

DEMO

Data Intention Language (DIL) - Use Case Domains

Domains

Standardize

Built-ins

Common vals

ORDER_ID	PRODUCT_ID	TOTAL_PAID	CURRENCY_CODE
3	3	8.99	EUR
2	2	8.99	GBP
1	1	8.99	NA
5	5	9.99	USD
4	4	1399.00	JPY
	Total	1435.96	

Need standard currency (EUR)

ORDER_ID	PRODUCT_ID	TOTAL_PAID	CURRENCY_CODE
3	3	8.99	EUR
2	2	8.99	GBP
1	1	8.99	NA
5	5	9.99	USD
4	4	1399.00	JPY
	Total	1435.96	

2 char currency?!

Mandatory

Amount
Currency
EUR ex rate

Codes

3 uppercase letters

Normalize

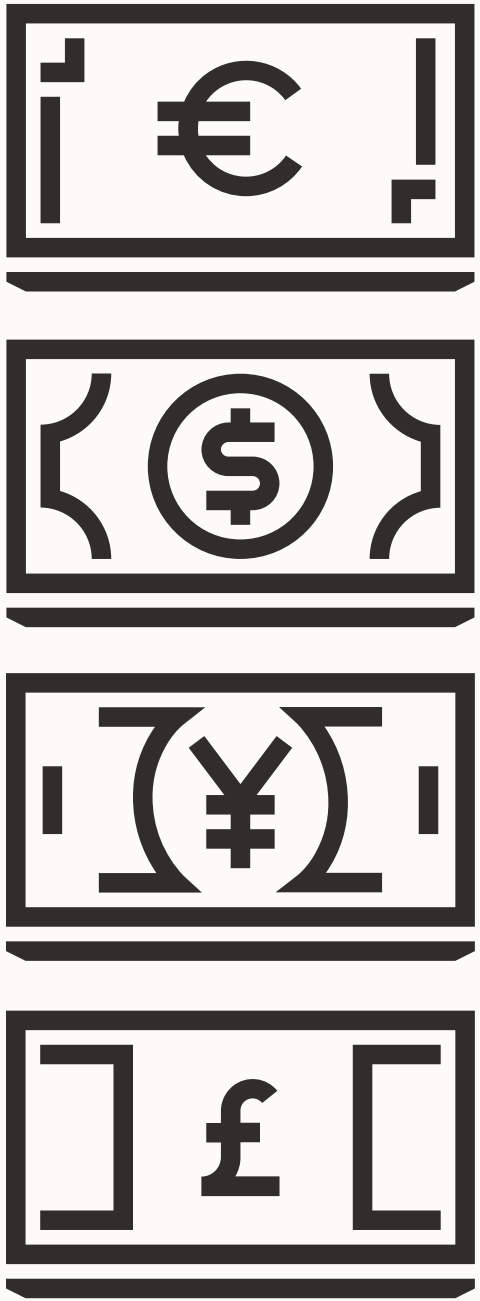
Convert to EUR

Multicolumn Domains

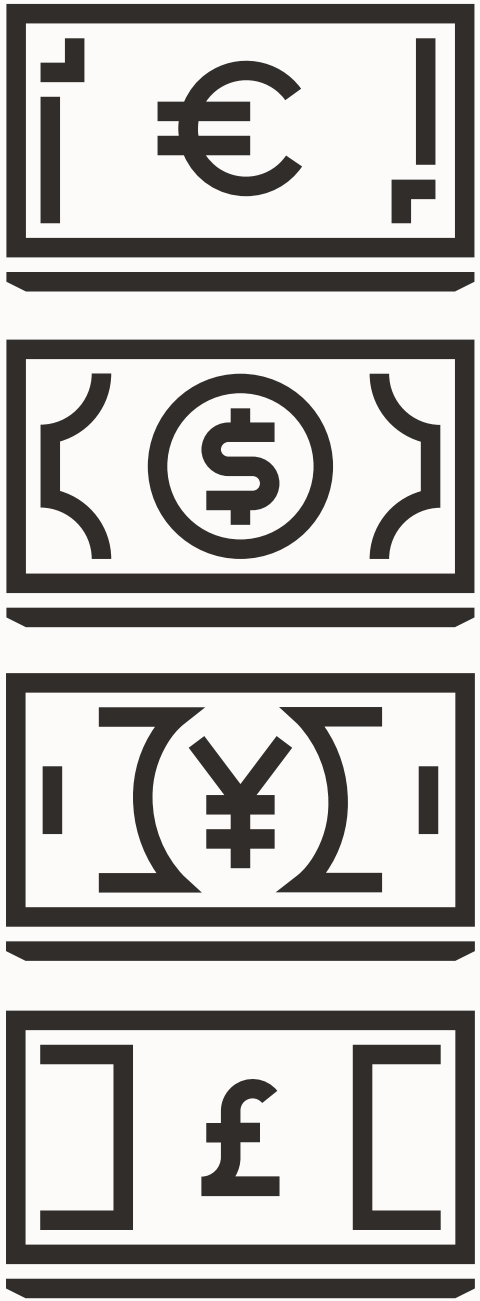
Describe composite values

```
create domain <multicolumn_domain_name> as (
  <column1> as <datatype> [ properties ],
  <column2> as <datatype> [ properties ],
  <column3> as <datatype> [ properties ],
  …
)
```

```
create domain currency as (
  amount               as number(10,2)
    not null,
  iso_currency_code as char(3 char) strict
    constraint curr_code_3_chars_c
    check ( regexp_like ( iso_currency_code, '[A-Z]{3}' ) )
    not null,
  exchange_rate      as number
    not null
)
```

Amount/price (number(10,2))
ISO currency code (3 chars)
Exchange rate (number)



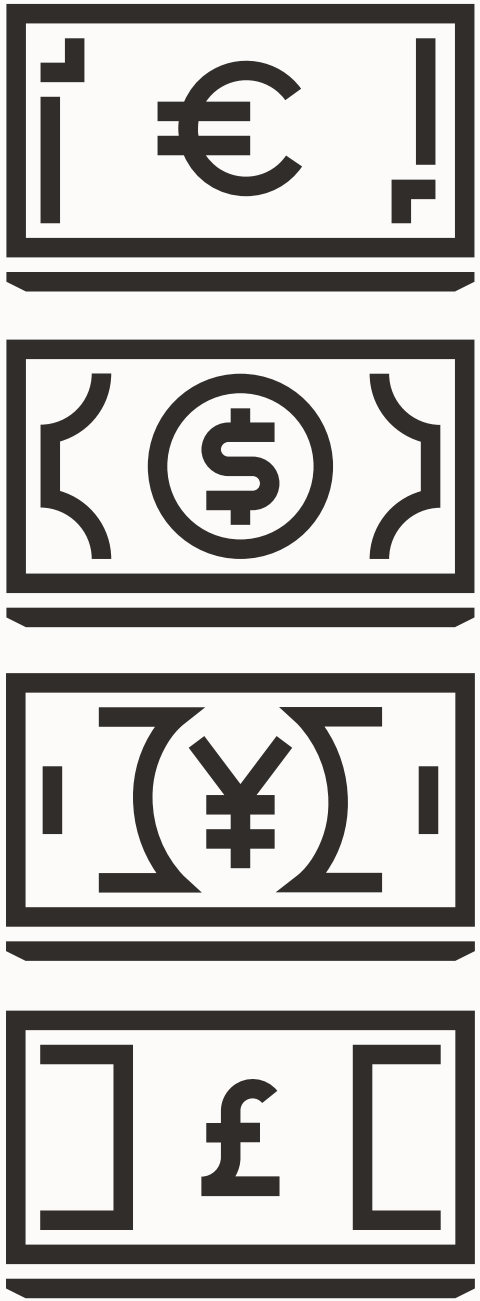
Varying units for monetary values;

€1.99 or 199¢?

Amount/price (number(10,2))

ISO currency code (3 chars)

Exchange rate (number)

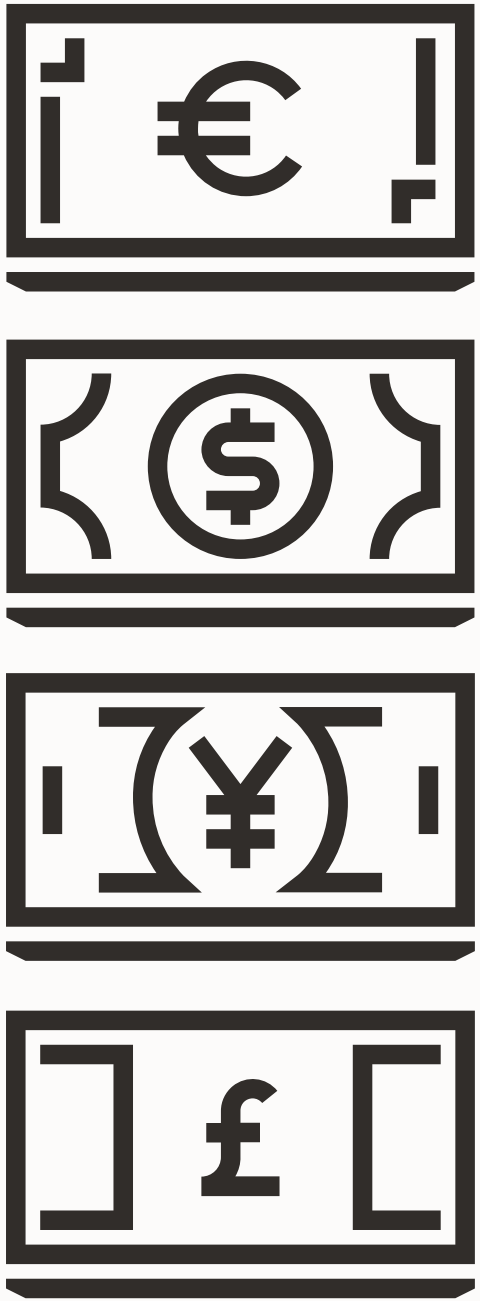


Amount/price (number(10,2))

ISO currency code (3 chars)

Exchange rate (number)

Match source API



Amount/price (number(10,2))

ISO currency code (3 chars)

Exchange rate (number)

Must be 3 characters

`strict` => exact match

```
create table order_items (
   …,
   total_paid          integer,
   currency_code       char (3 char),
   eur_exchange_rate   number(*,6),
   domain currency (
      total_paid, currency_code, usd_exchange_rate
   )
);
```

Precision & scale **ignored** for non-strict

```
create table invoice_items (
   …,
   total_paid          number(10, 2),
   currency_code       char (99 char),
   eur_exchange_rate number,
   domain currency (
      total_paid, currency_code, usd_exchange_rate
   )
);
```

Length mismatch for strict => **error**

```
create table invoice_items (
  …,
  total_paid          number(10, 2),
  currency_code       char (3 char),
  eur_exchange_rate number,
  domain currency (
    total_paid, currency_code, usd_exchange_rate
  )
);
```


```
alter table order_items
  modify (
    unit_price, currency_code, eur_ex_rate
  )
  add domain currency;
```

```
ORA-02293: cannot validate (%) - check constraint violated
```


```
create domain currency as (
  amount               as number(10,2)
    not null,
  iso_currency_code as char(3 char) strict
    constraint curr_code_3_chars_c
    check ( regexp_like ( iso_currency_code, '[A-Z]{3}' ) )
    novalidate
    not null,
  exchange_rate      as number
    not null
)
```


Ex rate formulas

Chris

```
alter table order_items
add ( amount_in_eur as (
  unit_price * ex_rate
) );
```

Alex

```
alter table invoice_items
add ( amount_in_eur as (
  amount / ex_rate
) );
```


Domains Functions

Rules for displaying & ordering values

```
alter domain currency
  add order amount * exchange_rate;

alter domain currency
  add display '(' || case iso_currency_code
    when 'NA' then 'XXX'
    else  iso_currency_code
  end || ')' || round ( amount * exchange_rate, 2 );
```

```
select order_id, product_id, total_paid,
       domain_display (
         total_paid, currency_code, usd_exchange_rate
       ) eur_amount
from   order_items
order  by domain_order (
  total_paid, currency_code, usd_exchange_rate
);
```

Call domain functions to use **sort** & **display** rules

ORDER_ID	PRODUCT_ID	TOTAL_PAID	EUR_AMOUNT
4	4	1399 (JPY)	8.46
2	2	8.99 (GBP)	8.54
3	3	8.99 (EUR)	8.99
5	5	8.99 (XXX)	8.99
1	1	9.99 (USD)	11.49
			46.47

```
alter table order_items add (
   amount_in_eur_display as ( domain_display (
      total_paid, currency_code, eur_exchange_rate
   ) ),
   amount_in_eur_order as ( domain_order (
      total_paid, currency_code, eur_exchange_rate
   ) )
);
```

DEMO

Data Intention Language (DIL) - Use Case Domains

Types

Single
Multicolumn
Enum
Flexible

Properties

Annotations
Constraints
Functions

Built-ins

Common values

Say what you want an app to do

Add schema intent

AI generates code

Data Intention Language (DIL)

Annotations

Key-value pairs
Application metadata

Use Case Domains

Extend base types
Describe data use case

Data Intention Language (DIL)

Consistent

Define rules once

Discovery

Info in data dictionary

Prompt

Augment LLM requests

NL.OUG
NL ORACLE USER GROUP

Fixing a Schema with Domains and Annotations-Chris Saxon & Alex Nuijten

Please fill in your evaluations