



# UNIVERSIDADE ESTADUAL DE SANTA CRUZ (UESC)

Criada pela Lei 6.344, de 05.12.1991,  
e reorganizada pela Lei 6.898, de 18.08.1995 e  
pela Lei 7.176, de 10.09.1997

## CET091 – Banco de Dados II

Prof. Dr. Marcelo Ossamu Honda

Departamento de Ciências Exatas e Tecnológicas (DCET)  
mohonda(at)uesc(.)br

# Bancos de Dados Baseados em Objeto

# Bancos de Dados Baseados em Objeto

- Aplicações tradicionais;
  - Baseadas no modelo de dados relacional;
- Bancos de dados baseados em objeto;
  - Modelo de dados relacional de objeto;
  - Estende o modelo de dados relacional;
    - Utilizar tipos de dados complexos e orientação a objetos;
    - Tentam preservar as fundações relacionais;
      - Estenda a capacidade de modelagem;
      - Preserva o acesso declarativo aos dados;
  - Dificuldade de acessar dados a partir de linguagens Orientadas a Objetos;

# Tipos de Dados Complexos

- Capaz de representar diretamente conceitos do modelo Entidade Relacionamento e Entidade Relacionamento Estendido;
  - Atributos compostos;
  - Atributos multivalorados;
  - Generalização;
  - Especialização;
- **Cuidadoso na aplicação dos tipos de dados complexos;**
  - **Como: conjuntos e Arrays;**

# Tipos de Dados Complexos

- 1 Forma Norma:
  - Todos os atributos tenham domínio atômico;
  - Consultas podem ser mais fáceis;

---

<i>title</i>	<i>author-set</i>	<i>publisher</i>	<i>keyword-set</i>
		( <i>name, branch</i> )	
Compilers	{Smith, Jones}	(McGraw-Hill, New York)	{parsing, analysis}
Networks	{Jones, Frick}	(Oxford, London)	{Internet, Web}

# Tipos de Dados Complexos

- 4 Forma Normal;
  - Exige que as consultas usem junções de várias relações;

<i>title</i>	<i>pub-name</i>	<i>pub-branch</i>
Compilers	McGraw-Hill	New York
Networks	Oxford	London

*books4*

<i>title</i>	<i>keyword</i>
Compilers	parsing
Compilers	analysis
Networks	Internet
Networks	Web

*keywords*

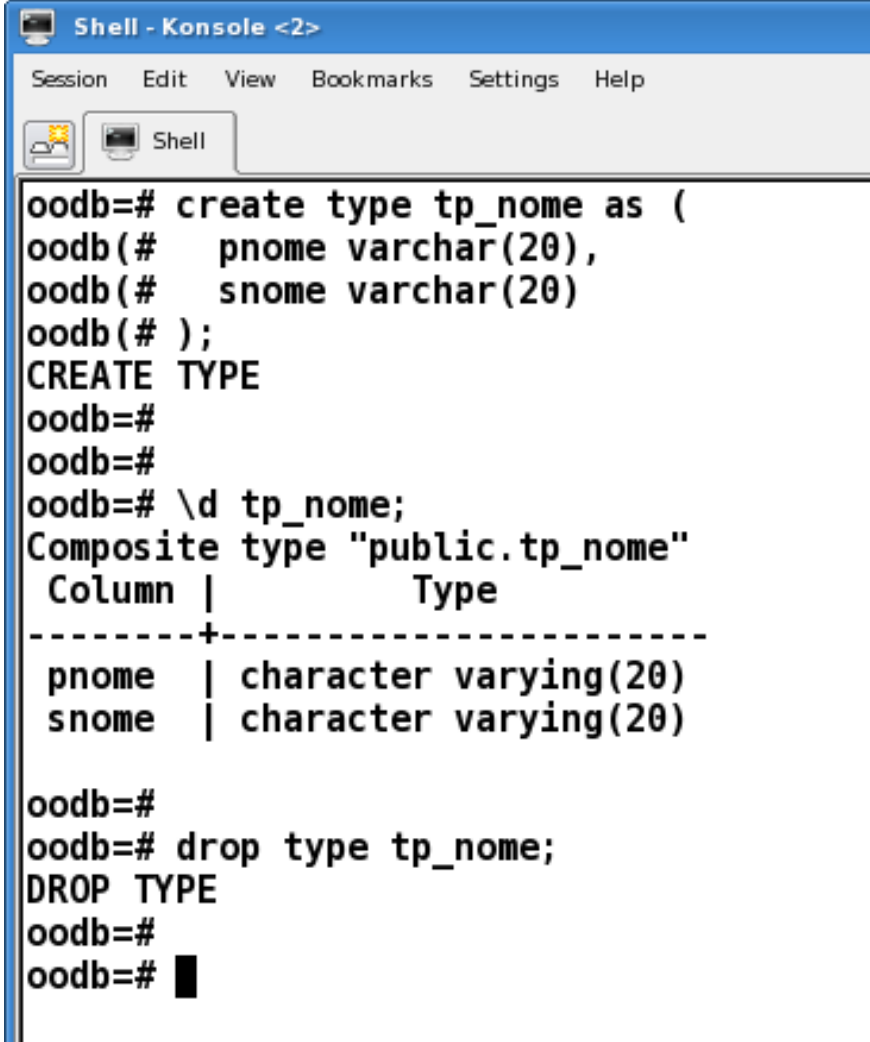
<i>title</i>	<i>author</i>
Compilers	Smith
Compilers	Jones
Networks	Jones
Networks	Frick

*authors*

# Tipos Estruturados

# Tipos Estruturados

- Permite representar atributos compostos;
  - Tipos definidos pelo usuário;
- Exemplo;
  - Define um tipo composto;
  - Consulta a estrutura do tipo definido composto;



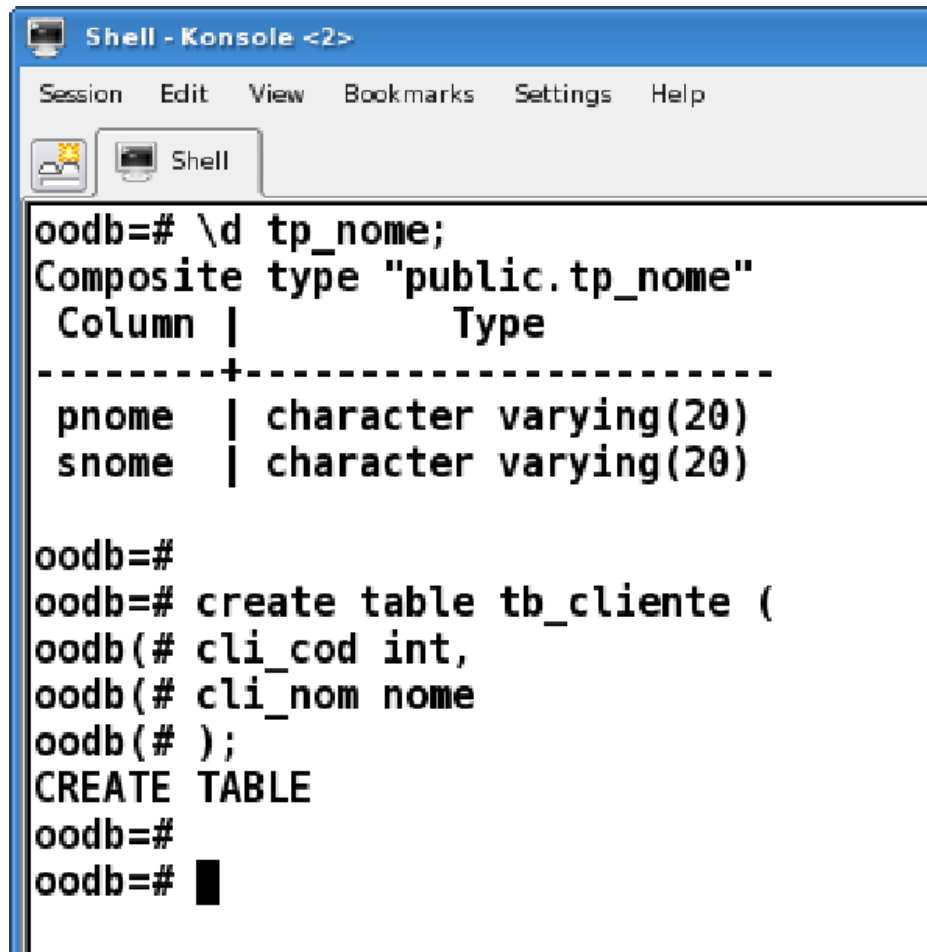
```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help

oodb=# create type tp_nome as (
oodb(#   pnome varchar(20),
oodb(#   snome varchar(20)
oodb(# );
CREATE TYPE
oodb=#
oodb=#
oodb=# \d tp_nome;
Composite type "public.tp_nome"
Column |          Type
-----+-----
pnome   | character varying(20)
snome   | character varying(20)

oodb=#
oodb=# drop type tp_nome;
DROP TYPE
oodb=#
oodb=# █
```



# Tipos Estruturados



```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help
Shell

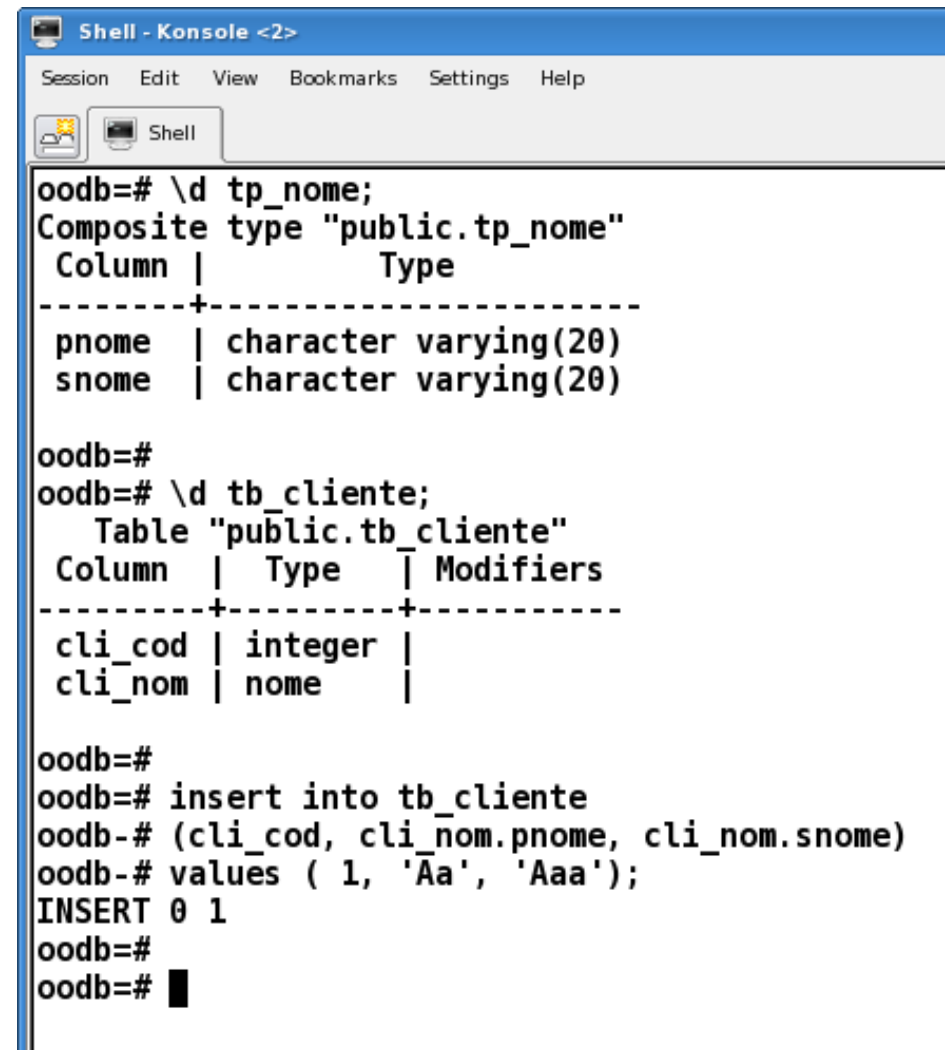
oodb=# \d tp_nome;
Composite type "public.tp_nome"
Column |          Type
-----+-----
pnome  | character varying(20)
snome  | character varying(20)

oodb=#
oodb=# create table tb_cliente (
oodb(# cli_cod int,
oodb(# cli_nom nome
oodb(# );
CREATE TABLE
oodb=#
oodb=# █
```

- Exemplo;
  - Cria uma tabela utilizando um tipo definido composto;

# Tipos Estruturados

- Exemplo;
  - Insere dados em um tabela com um tipo definido composto;



```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help

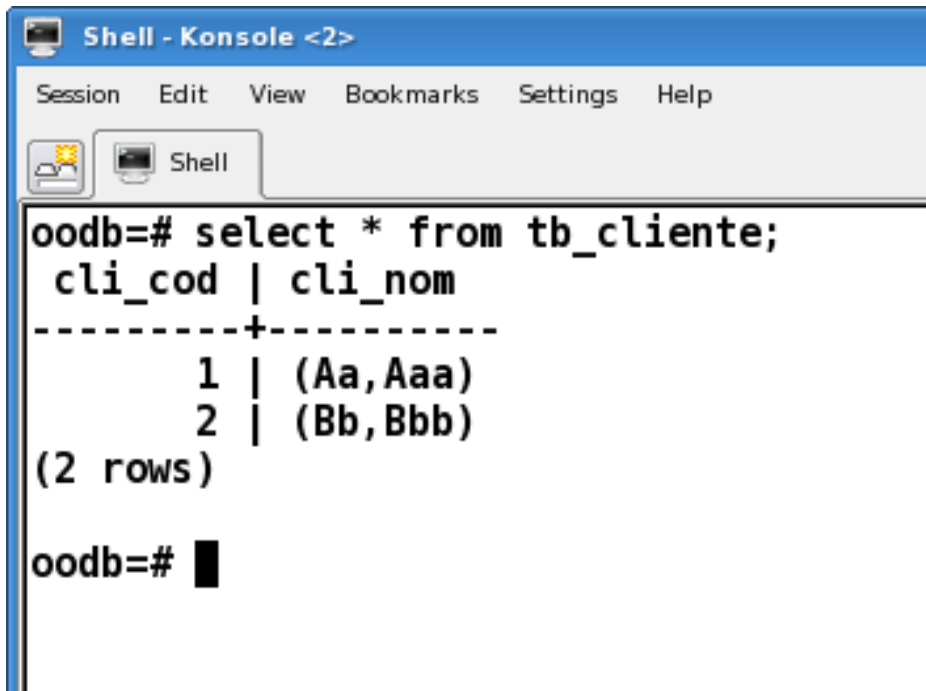
oodb=# \d tp_nome;
Composite type "public.tp_nome"
Column |          Type
-----+-----
pname  | character varying(20)
sname  | character varying(20)

oodb=#
oodb=# \d tb_cliente;
Table "public.tb_cliente"
Column | Type      | Modifiers
-----+-----+-----
cli_cod | integer   |
cli_nom | nome      |

oodb=#
oodb=# insert into tb_cliente
oodb=# (cli_cod, cli_nom.pnome, cli_nom.snome)
oodb=# values ( 1, 'Aa', 'Aaa');
INSERT 0 1
oodb=#
oodb=#
```

# Tipos Estruturados

- Exemplo;
  - Consulta (*simples*) de dados em um tabela com um tipo definido composto;



The screenshot shows a terminal window titled "Shell - Konsole <2>". The window has a menu bar with "Session", "Edit", "View", "Bookmarks", "Settings", and "Help". Below the menu bar is a tab labeled "Shell". The terminal content shows a SQL query being executed: "oodb=# select \* from tb\_cliente;". The output is a table with two columns: "cli\_cod" and "cli\_nom". The table has two rows of data: (1, (Aa, Aaa)) and (2, (Bb, Bbb)). The output is formatted with a header row, a separator row, and the data rows. Below the data rows, it says "(2 rows)". The prompt "oodb=#" is followed by a cursor.

```
oodb=# select * from tb_cliente;
cli_cod | cli_nom
-----+-----
      1 | (Aa,Aaa)
      2 | (Bb,Bbb)
(2 rows)

oodb=#
```

# Tipos Estruturados

- Exemplo;
  - Consulta utilizando um tipo definido composto de uma tabela;
    - Atenção as regras do PostgreSQL;

```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help

oodb=# \d tb_cliente;
Table "public.tb_cliente"
Column | Type      | Modifiers
-----+-----+-----
cli_cod | integer   |
cli_nom | nome      |

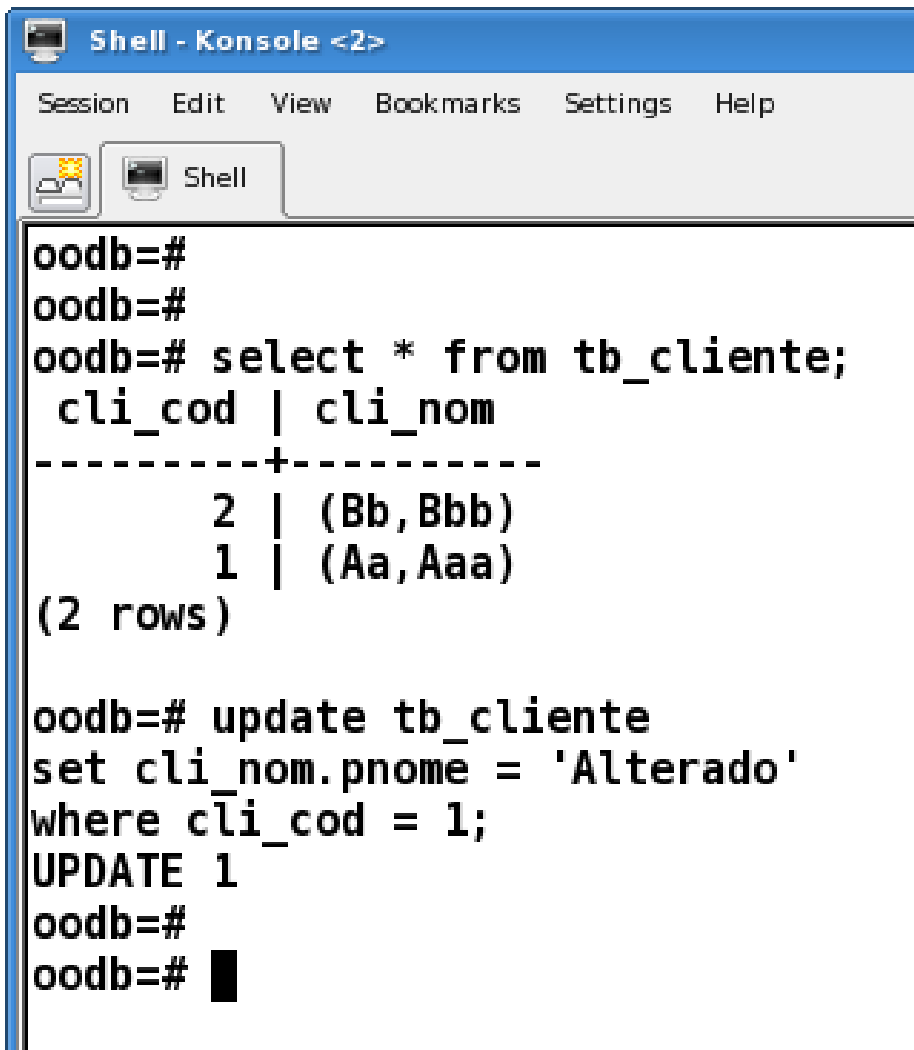
oodb=#
oodb=# \d tp_nome;
Composite type "public.tp_nome"
Column | Type
-----+-----
pnome  | character varying(20)
snome  | character varying(20)

oodb=#
oodb=# select cli_cod, (tb_cliente).cli_nom.pnome
oodb=# from tb_cliente;
 cli_cod | pnome
-----+-----
        1 | Aa
        2 | Bb
(2 rows)

oodb=#
```

# Tipos Estruturados

- Exemplo;
  - Atualização de um tipo definido composto de uma tabela;



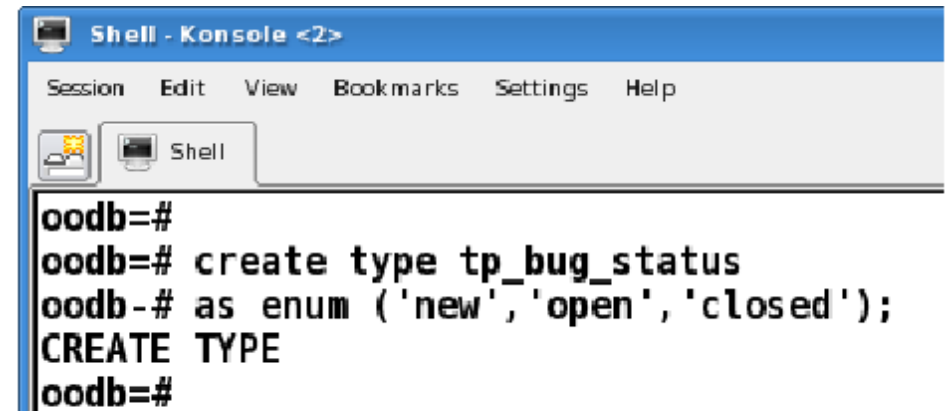
```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help

oodb=#
oodb=#
oodb=# select * from tb_cliente;
 cli_cod | cli_nom
-----+-----
         2 | (Bb,Bbb)
         1 | (Aa,Aaa)
(2 rows)

oodb=# update tb_cliente
set cli_nom.pnome = 'Alterado'
where cli_cod = 1;
UPDATE 1
oodb=#
oodb=#
```

# Tipos Estruturados

- Tipo ENUM;
- Exemplo;
  - Consulta aos dados de um tipo definido ENUM;



```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help
oodb=#
oodb=# create type tp_bug_status
oodb-# as enum ('new','open','closed');
CREATE TYPE
oodb=#
```

# Tipos Estruturados

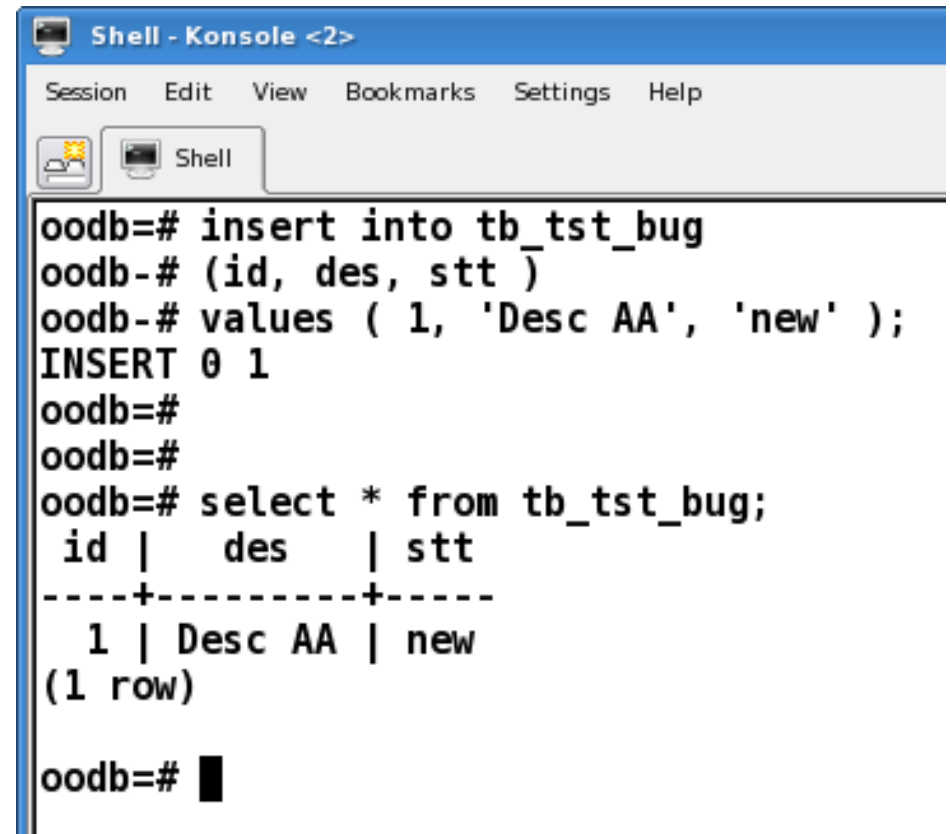
- Exemplo;
  - Criando uma tabela usando um tipo definido ENUM;

```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help
Shell
oodb=# create table tb_tst_bug (
oodb(# id int primary key,
oodb(# des text,
oodb(# stt tp_bug_status
oodb(# );
NOTICE: CREATE TABLE / PRIMARY KE'
le "tb_tst_bug"
CREATE TABLE
oodb=#
```

```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help
Shell
oodb=# select enum_range(null::tp_bug_status)
oodb-# from tb_tst_bug;
enum_range
-----
{new, open, closed}
(1 row)
oodb=#
```

# Tipos Estruturados

- Exemplo;
  - Inserindo dados em uma tabela usando um tipo definido ENUM;



```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help

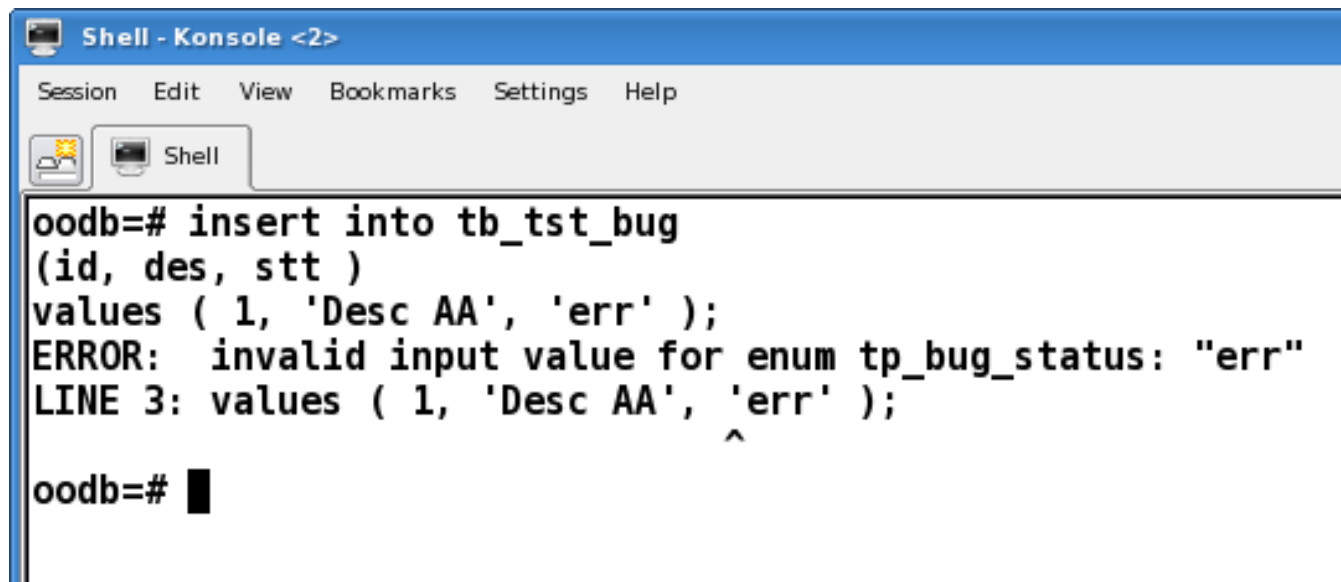
oodb=# insert into tb_tst_bug
oodb-# (id, des, stt )
oodb-# values ( 1, 'Desc AA', 'new' );
INSERT 0 1
oodb=#
oodb=#
oodb=# select * from tb_tst_bug;
 id |  des   | stt
----+-----+----
  1 | Desc AA | new
(1 row)

oodb=#
```



# Tipos Estruturados

- Exemplo;
  - Inserindo dados (errados) em uma tabela usando um tipo definido ENUM;



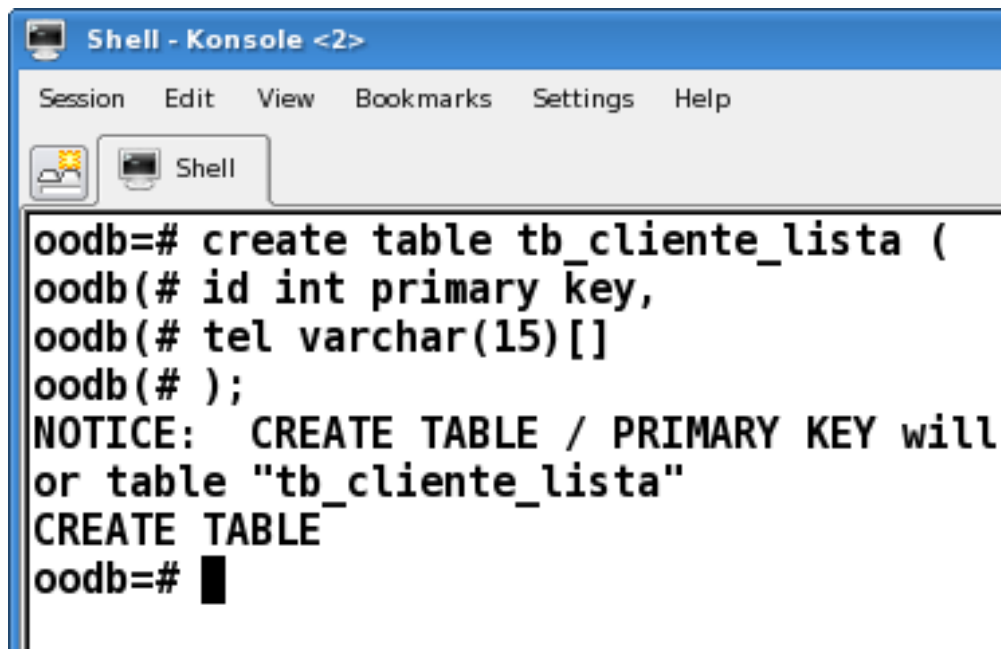
```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help

oodb=# insert into tb_tst_bug
(id, des, stt )
values ( 1, 'Desc AA', 'err' );
ERROR:  invalid input value for enum tp_bug_status: "err"
LINE 3: values ( 1, 'Desc AA', 'err' );
                                   ^

oodb=#
```

# Tipos Estruturados

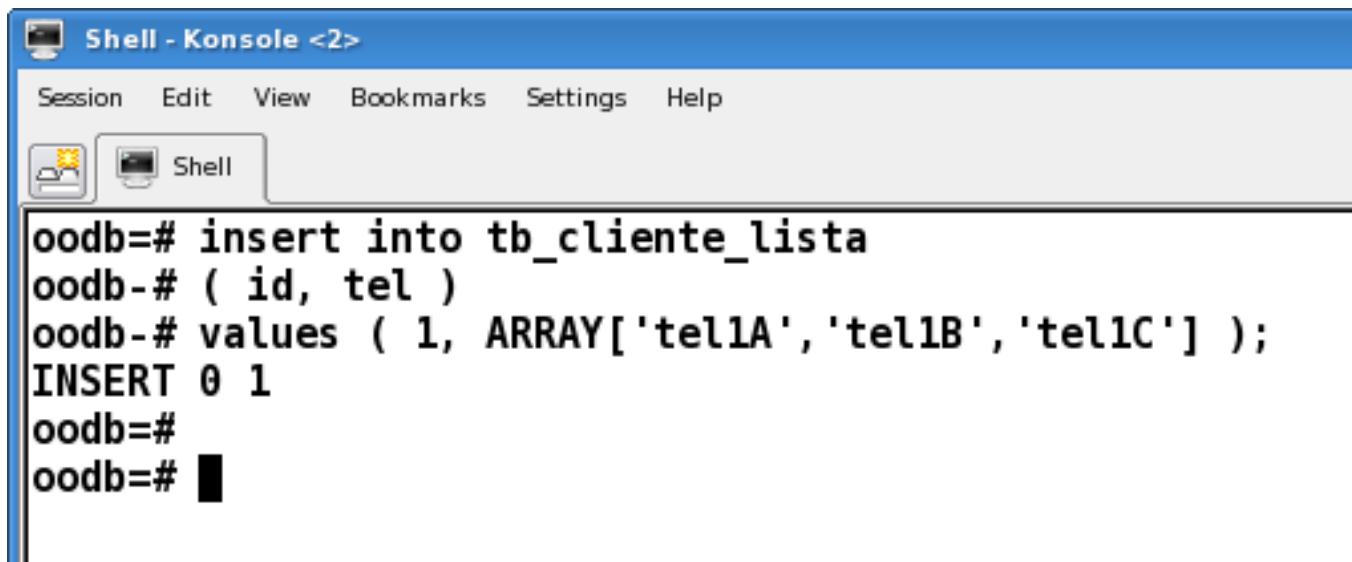
- Tipo Arrays;
- Exemplo:
  - Lista indefinida de elementos;



```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help
Shell
oodb=# create table tb_cliente_lista (
oodb(# id int primary key,
oodb(# tel varchar(15)[]
oodb(# );
NOTICE: CREATE TABLE / PRIMARY KEY will
or table "tb_cliente_lista"
CREATE TABLE
oodb=#
```

# Tipos Estruturados

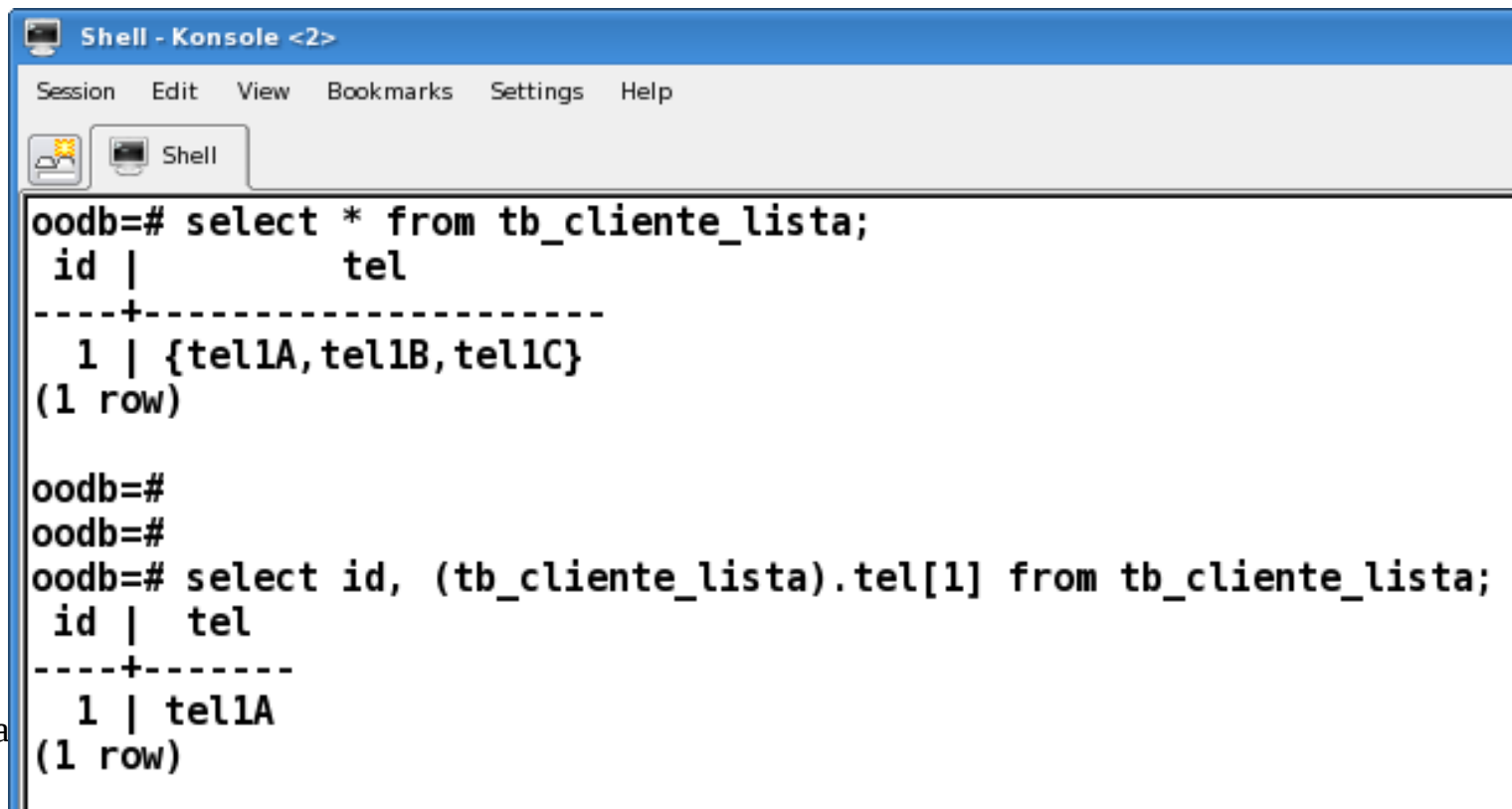
- Exemplo;
  - Inserindo dados em um array;



```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help
Shell
oodb=# insert into tb_cliente_lista
oodb-# ( id, tel )
oodb-# values ( 1, ARRAY['tel1A','tel1B','tel1C'] );
INSERT 0 1
oodb=#
oodb=#
```

# Tipos Estruturados

- Exemplo;
  - Consultando dados de um array;
    - Atenção as regras do PostgreSQL;



```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help

oodb=# select * from tb_cliente_lista;
 id |          tel
-----+-----
  1 | {tel1A,tel1B,tel1C}
(1 row)

oodb=#
oodb=#
oodb=# select id, (tb_cliente_lista).tel[1] from tb_cliente_lista;
 id | tel
-----+-----
  1 | tel1A
(1 row)
```

# Herança

# Herança

- Também definida como subtabelas;
- Correspondem a noção do modelo entidade relacionamento;
  - Generalização;
  - Especialização;

# Herança

- Exemplos;
- Definindo tipos de dados;

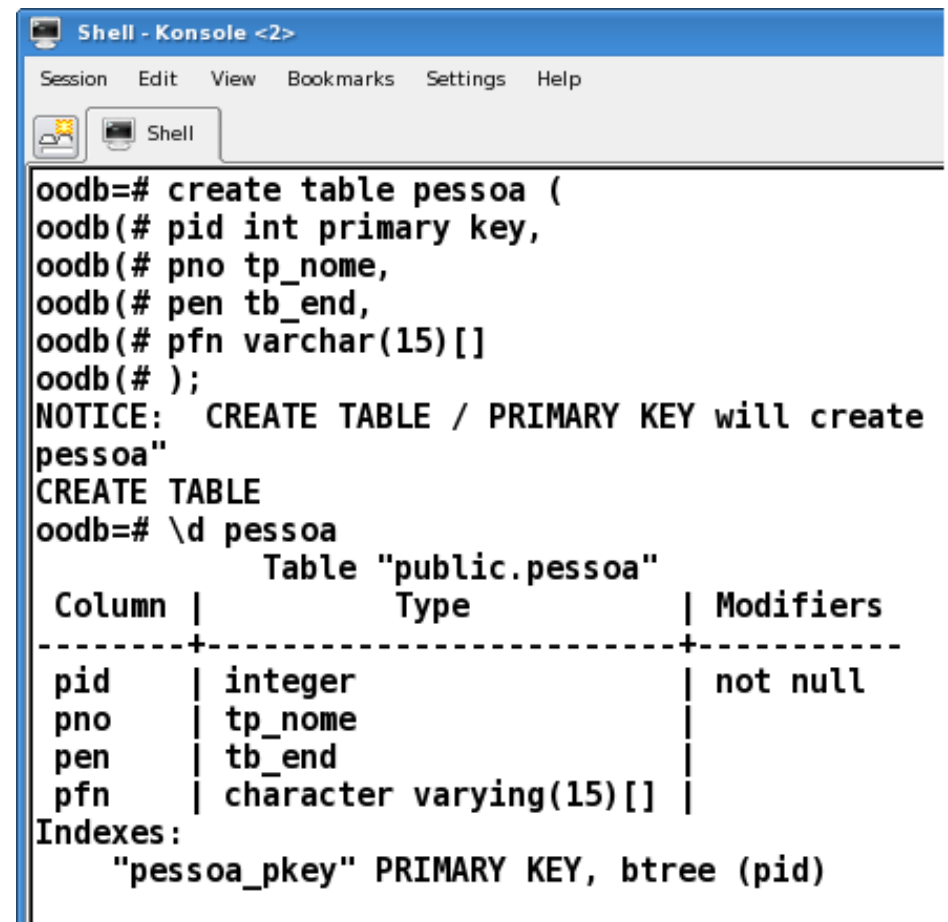
```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help
Shell
oodb=# create type tp_nome as (
oodb(#  pnome varchar(20),
oodb(#  snome varchar(20)
oodb(# );
CREATE TYPE
oodb=#
oodb=#
```

```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help
Shell
oodb=# create type tp_estados
oodb-# as enum ( 'Acre', 'Alagoas', 'Amapá', 'Amazonas',
oodb(# 'Bahia', 'Ceará', 'Distrito Federal', 'Espírito Santo',
oodb(# 'Goiás', 'Maranhão', 'Mato Grosso', 'Mato Grosso do Sul',
oodb(# 'Minas Gerais', 'Pará', 'Paraíba', 'Paraná', 'Pernambuco',
oodb(# 'Piauí', 'Rio de Janeiro', 'Rio Grande do Norte',
oodb(# 'Rio Grande do Sul', 'Rondônia', 'Roraima',
oodb(# 'Santa Catarina', 'São Paulo', 'Sergipe',
oodb(# 'Tocantins'
oodb(# );
CREATE TYPE
oodb=#
```

```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help
Shell
oodb=# create type tb_end as (
oodb(# rua varchar(75),
oodb(# numero smallint,
oodb(# bairro varchar(40),
oodb(# cidade varchar(50),
oodb(# cep char(8),
oodb(# estado tp_estados,
oodb(# complemento varchar(30)
oodb(# );
CREATE TYPE
oodb=#
```

# Herança

- Exemplo;
  - Criando a tabela pessoa;
    - Será usada como tabela pai;
  - Verificando a estrutura da tabela pessoa;

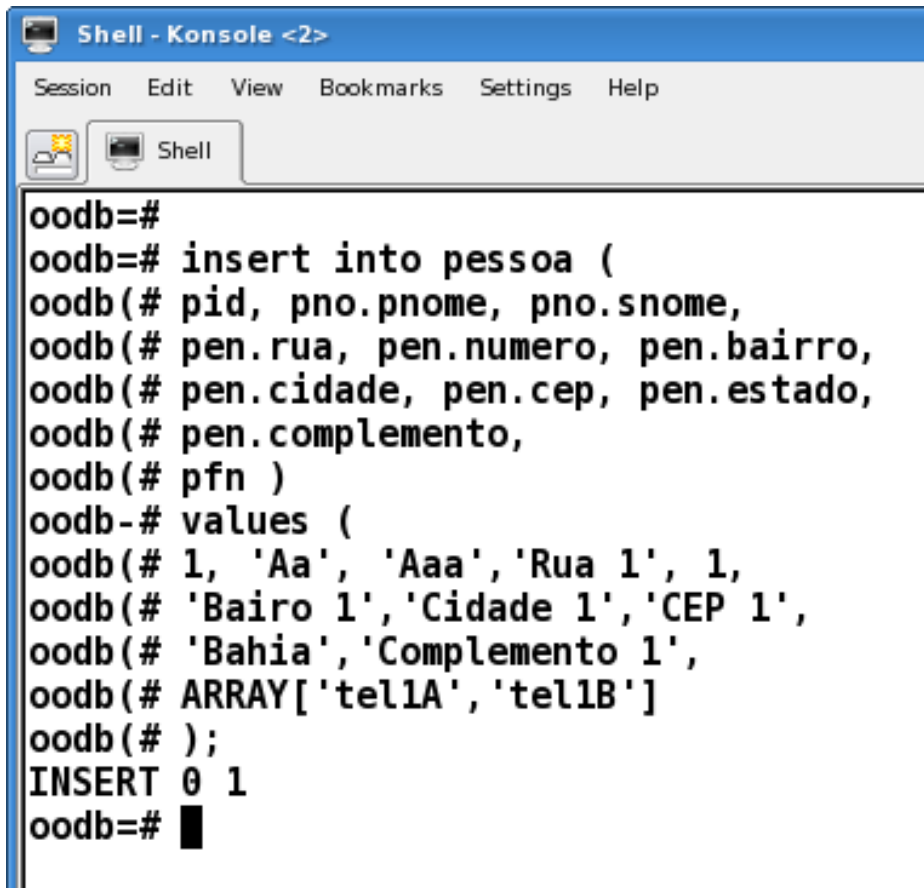


```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help
Shell
oodb=# create table pessoa (
oodb(# pid int primary key,
oodb(# pno tp_nome,
oodb(# pen tb_end,
oodb(# pfn varchar(15)[]
oodb(# );
NOTICE: CREATE TABLE / PRIMARY KEY will create
pessoa"
CREATE TABLE
oodb=# \d pessoa
Table "public.pessoa"
Column | Type | Modifiers
-----+-----+-----
pid | integer | not null
pno | tp_nome |
pen | tb_end |
pfn | character varying(15)[] |
Indexes:
    "pessoa_pkey" PRIMARY KEY, btree (pid)
```



# Herança

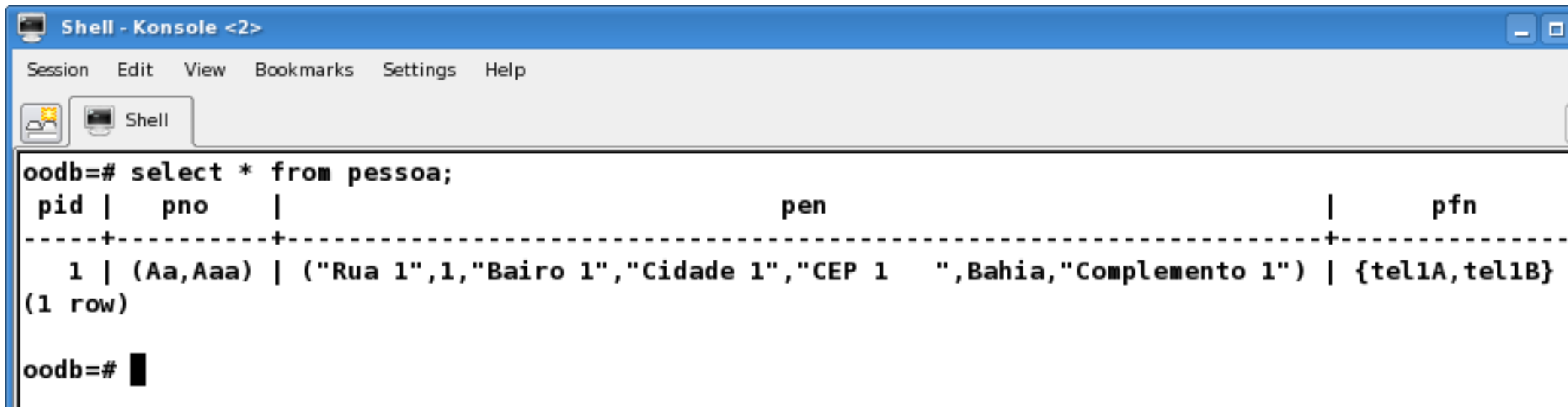
- Exemplo;
  - Inserindo dados na tabela pessoa;

A screenshot of a terminal window titled "Shell - Konsole <2>". The window has a menu bar with "Session", "Edit", "View", "Bookmarks", "Settings", and "Help". Below the menu bar is a tab labeled "Shell". The terminal content shows an SQL insert command being executed in a shell prompt "oodb=#". The command is: "insert into pessoa ( pid, pno.pnome, pno.snome, pen.rua, pen.numero, pen.bairro, pen.cidade, pen.cep, pen.estado, pen.complemento, pfn ) values ( 1, 'Aa', 'Aaa', 'Rua 1', 1, 'Bairro 1', 'Cidade 1', 'CEP 1', 'Bahia', 'Complemento 1', ARRAY['tel1A', 'tel1B'] );". The output of the command is "INSERT 0 1". The prompt "oodb=#" is followed by a cursor.

```
oodb=#
oodb=# insert into pessoa (
oodb(# pid, pno.pnome, pno.snome,
oodb(# pen.rua, pen.numero, pen.bairro,
oodb(# pen.cidade, pen.cep, pen.estado,
oodb(# pen.complemento,
oodb(# pfn )
oodb-# values (
oodb(# 1, 'Aa', 'Aaa', 'Rua 1', 1,
oodb(# 'Bairro 1', 'Cidade 1', 'CEP 1',
oodb(# 'Bahia', 'Complemento 1',
oodb(# ARRAY['tel1A', 'tel1B']
oodb(# );
INSERT 0 1
oodb=# █
```

# Herança

- Exemplo;
  - Consultando dados na tabela pessoa;



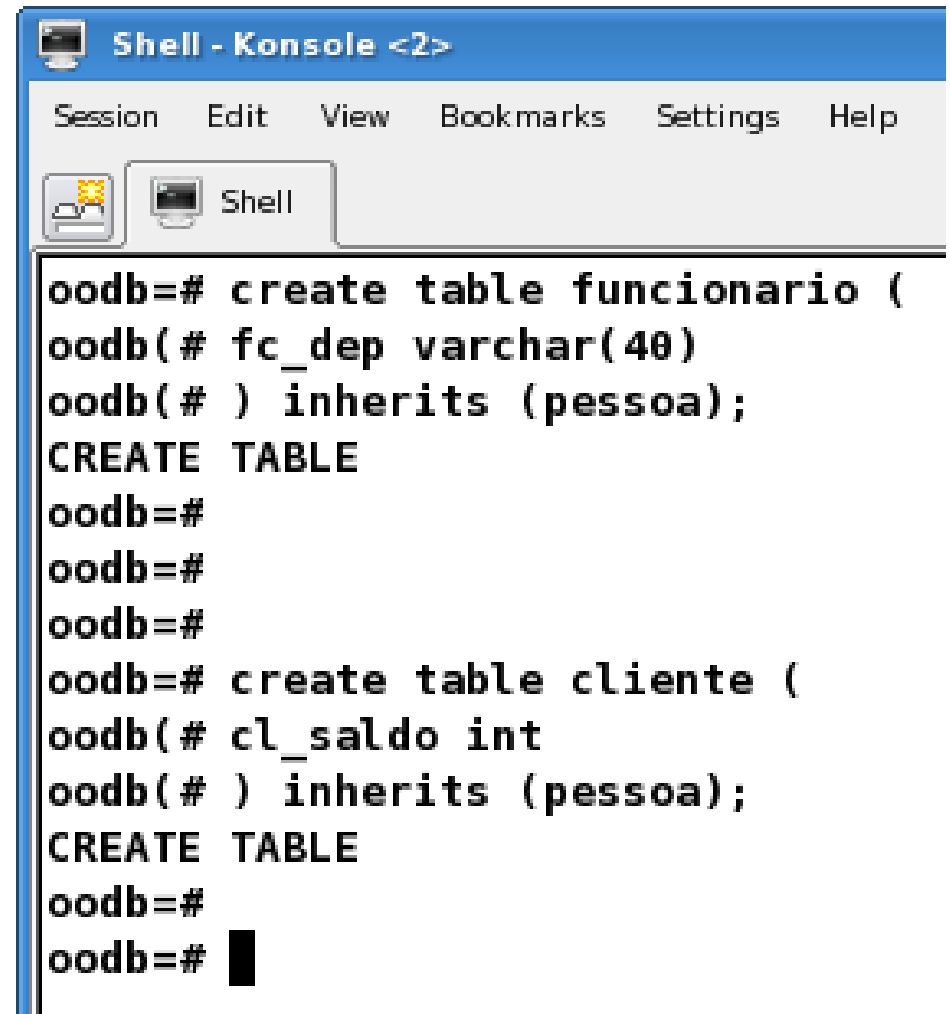
```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help

oodb=# select * from pessoa;
 pid | pno | pen | pfn
-----+-----+-----+-----
  1 | (Aa,Aaa) | ("Rua 1",1,"Bairro 1","Cidade 1","CEP 1",Bahia,"Complemento 1") | {tel1A,tel1B}
(1 row)

oodb=#
```

# Herança

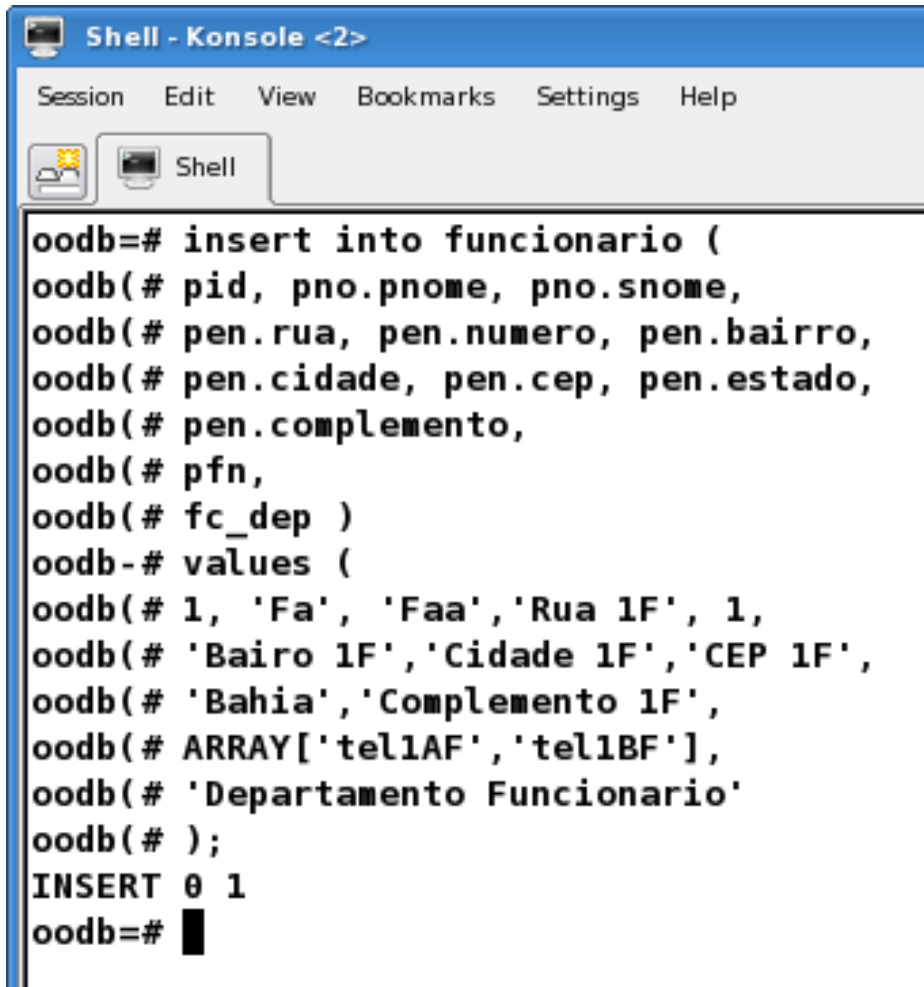
- Exemplo;
  - Criando as tabelas filhas;



```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help

oodb=# create table funcionario (
oodb(# fc_dep varchar(40)
oodb(# ) inherits (pessoa);
CREATE TABLE
oodb=#
oodb=#
oodb=#
oodb=# create table cliente (
oodb(# cl_saldo int
oodb(# ) inherits (pessoa);
CREATE TABLE
oodb=#
oodb=#
```

# Herança



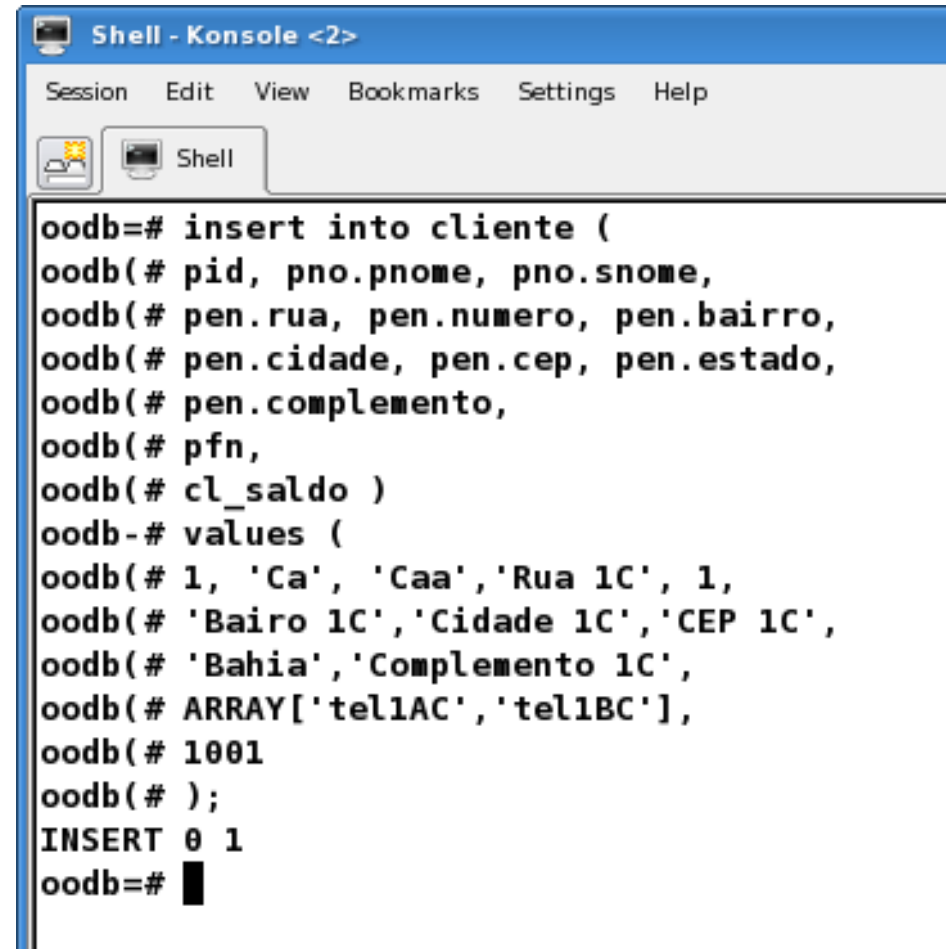
```
Shell - Konsole <2>
Session Edit View Bookmarks Settings Help

oodb=# insert into funcionario (
oodb(# pid, pno.pnome, pno.snome,
oodb(# pen.rua, pen.numero, pen.bairro,
oodb(# pen.cidade, pen.cep, pen.estado,
oodb(# pen.complemento,
oodb(# pfn,
oodb(# fc_dep )
oodb-# values (
oodb(# 1, 'Fa', 'Faa','Rua 1F', 1,
oodb(# 'Bairro 1F','Cidade 1F','CEP 1F',
oodb(# 'Bahia','Complemento 1F',
oodb(# ARRAY['tel1AF','tel1BF'],
oodb(# 'Departamento Funcionario'
oodb(# );
INSERT 0 1
oodb=#
```

- Exemplo;
  - Inserindo dados na tabela funcionário;

# Herança

- Exemplo;
  - Inserindo dados na tabela cliente;

A screenshot of a terminal window titled "Shell - Konsole <2>". The window has a menu bar with "Session", "Edit", "View", "Bookmarks", "Settings", and "Help". Below the menu bar is a toolbar with icons for a file, a printer, and a "Shell" button. The main area of the terminal displays an SQL insert command in a monospaced font. The command is: 

```
oodb=# insert into cliente (  
oodb(# pid, pno.pnome, pno.snome,  
oodb(# pen.rua, pen.numero, pen.bairro,  
oodb(# pen.cidade, pen.cep, pen.estado,  
oodb(# pen.complemento,  
oodb(# pfn,  
oodb(# cl_saldo )  
oodb-# values (  
oodb(# 1, 'Ca', 'Caa','Rua 1C', 1,  
oodb(# 'Bairro 1C','Cidade 1C','CEP 1C',  
oodb(# 'Bahia','Complemento 1C',  
oodb(# ARRAY['tel1AC','tel1BC'],  
oodb(# 1001  
oodb(# );  
INSERT 0 1  
oodb=#
```

# Herança

- Exemplos;
  - Consultas nas tabelas filhas;
  - Tabela pai, possui as tuplas de todas as outras tabelas;

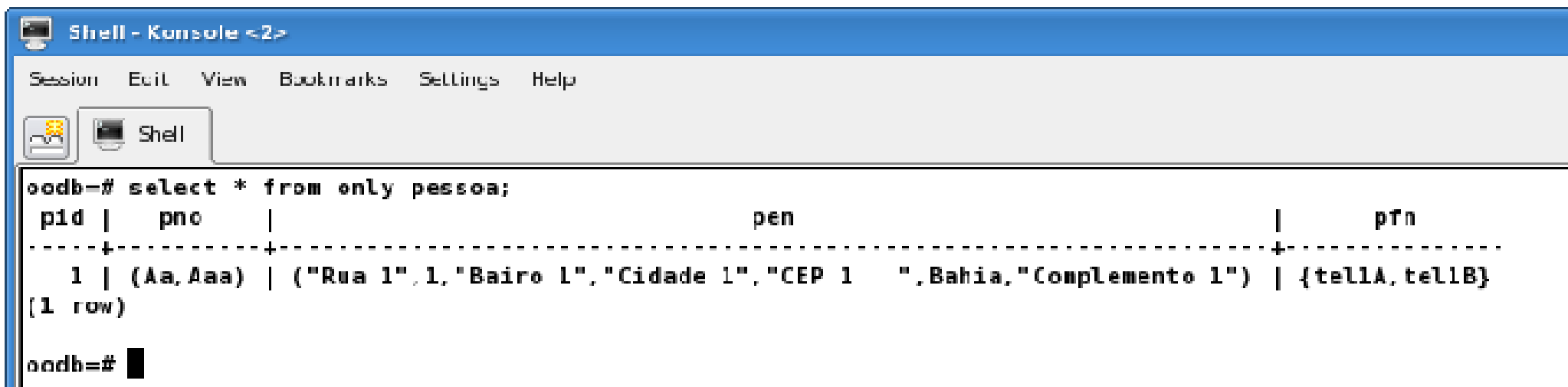
```
Shell - Konsole ~2>
Session Edit View Bookmarks Settings Help
Shell
oodb=# select * from cliente;
pid | pno | pen | pfn | cl_saldo
-----+-----+-----+-----+-----
1 | (Ca, Caa) | ("Rua 1C",1,"Bairro 1C","Cidade 1C","CEP 1C ",Bahia,"Complemento 1C") | {tel1AC,tel1BC} | 1001
[1 row]

oodb=#
oodb=# select * from funcionario;
pid | pno | pen | pfn | fc_dep
-----+-----+-----+-----+-----
1 | (Fa, Faa) | ("Rua 1F",1,"Bairro 1F","Cidade 1F","CEP 1F ",Bahia,"Complemento 1F") | {tel1AF,tel1BF} | Departamento Funcionario
[1 row]

oodb=#
oodb=# select * from pessoa:
pid | pno | pen | pfn
-----+-----+-----+-----
1 | (Aa, Aaa) | ("Rua 1",1,"Bairro 1","Cidade 1","CEP 1 ",Bahia,"Complemento 1") | {tel1A,tel1B}
1 | (Fa, Faa) | ("Rua 1F",1,"Bairro 1F","Cidade 1F","CEP 1F ",Bahia,"Complemento 1F") | {tel1AF,tel1BF}
1 | (Ca, Caa) | ("Rua 1C",1,"Bairro 1C","Cidade 1C","CEP 1C ",Bahia,"Complemento 1C") | {tel1AC,tel1BC}
[3 rows]
```

# Herança

- Exemplo;
  - Uso da cláusula **only**;
  - Lista somente as tuplas pertencentes a tabela pai;



The screenshot shows a terminal window titled "Shell - Konsole <2>". The terminal displays a SQL query and its result. The query is `oodb=# select * from only pessoa;`. The result is a table with columns `pid`, `pno`, `pen`, and `pfn`. The table contains one row with the following values: `1`, `(Aa,Aaa)`, `("Rua 1",1,"Bairro 1","Cidade 1","CEP 1", "Bahia,"(complemento 1"))`, and `{tel1A,tel1B}`. The terminal also shows `[1 row]` and `oodb=#` at the bottom.

```
oodb=# select * from only pessoa;
pid | pno | pen | pfn
-----+-----+-----+-----
  1 | (Aa,Aaa) | ("Rua 1",1,"Bairro 1","Cidade 1","CEP 1", "Bahia,"(complemento 1")) | {tel1A,tel1B}
[1 row]
oodb=#
```

# Referências

- Ramez Elmasri e Shamkant B, Navathe, Sistemas de Banco de Dados, Pearson Addison Wesley, 2005;
- Abraham Silverschatz, Henry F. Korth e S. Sudarshan, Sistema de Banco de Dados, Editora Campus, 2006;
- PostgreSQL 8.4.0 Documentation, by The PostgreSQL Global Development Group, Copyright © 1996-2009 The PostgreSQL Global Development Group;