

Para os exercícios abaixo, utilize a base de dados de venda fornecida pela professora (está no Moodle). Não se esqueça de postar as respostas no Moodle.

```
cliente (codigo, nome, email, telefone)
funcionario (codigo, nome, email, dtaNasc, salario)
venda (numero, data, hora, codclie#, codFun#, tipo)
      codClie REFERENCES cliente (codigo)
      codFun REFERENCES funcionario (codigo)
produto (codigo, nome, preco, qtdEstoque)
produtoVendido (numero#, codProd#, qtd, valor)
      numero REFERENCES venda (numero),
      codProd REFERENCES produto (codigo)
```

1. Retornar uma tabela com as colunas nomes e e-mails, de clientes e funcionários que são vendedores (apenas funcionário que aparece na tabela venda é considerado um vendedor). Deve-se criar uma terceira coluna que indique se ele é 'cliente' ou 'funcionário'.

nome character varying(40)	email character varying(40)	categoria text
Lobervau	lob@empresa.com	funcion
Aristideles	ari@empresa.com	funcion
Robercleberson	rob@empresa.com	funcion
Leinaura	lei@empresa.com	funcion
Berlindo	be@empresa.com	funcion
Claudison	clau@empresa.com	funcion
fluvialson	flu@empresa.com	funcion
Paulo Marcos	pama@abc.de.br	cliente
Paulinha Vera	pva@abc.de.br	cliente
Linda Regina	lire@abc.de.br	cliente
Tatiana Clara	tacl@abc.de.br	cliente
Juliana Paula	juju@abc.de.br	cliente
Julili	juli@abc.de.br	cliente
Flavio Augusto	flau@abc.de.br	cliente
Marco Antonio	mat@abc.de.br	cliente
Monira Rosa	moro@abc.de.br	cliente
Tamires Ana	tana@abc.de.br	cliente
Claudia Regina	cr@abc.de.br	cliente
Claudio Rogério	crog@abc.de.br	cliente
Luciano Paulo	lupa@abc.de.br	cliente
Robercleberson	rob@empresa.com	cliente

```
SELECT c.nome, c.email, 'Cliente' as categoria
FROM Cliente c
```



UNION

```
SELECT f.nome, f.email, 'Funcionario' as categoria
FROM Funcionario f JOIN Venda v ON v.codFun = f.codigo
ORDER BY 3 DESC
```

2. Execute a mesma consulta feita em 1), mas retirando a terceira coluna. O que mudou na resposta? Porque?
A execução gerou uma linha a menos que a outra, pois UNION não mostra dados repetidos.
3. Nome e email do funcionário que também é cliente

```
SELECT c.nome, c.email
FROM Cliente c
INTERSECT
SELECT f.nome, f.email
FROM Funcionario f
Order By 1
```

4. Código e nome dos funcionários e clientes que nunca apareceram em um venda.

codigo integer	nome character varying(40)
4	Norlindson
9	Jucemar

```
SELECT c.codigo, c.nome
FROM Cliente c
EXCEPT
SELECT DISTINCT c.codigo, c.nome
FROM Cliente c JOIN venda v ON v.codclie = c.codigo
```

UNION

```
SELECT f.codigo, f.nome
FROM Funcionario f
EXCEPT
SELECT f.codigo, f.nome
FROM Funcionario f JOIN venda v ON v.codfun = f.codigo
```

5. Nome e email dos clientes que realizaram compras do tipo “À VISTA”. Usar pelo menos uma vez um operador de conjunto.

nome character varying(40)	email character varying(40)
Flavio Augusto	flau@abc.de.br
Luciano Paulo	lupa@abc.de.br
Marco Antonio	mat@abc.de.br
Paulo Marcos	pama@abc.de.br
Tatiana Clara	tacl@abc.de.br
Robercleberson	rob@empresa.com
Paulinha Vera	pve@abc.de.br
Julili	juli@abc.de.br

6. Nomes dos produtos que não foram vendidos no período de: início de 2004 até final de 2007. Caso seja necessário mais de um SELECT, use operador de conjunto em algum deles.

nome character varying(40)
Chapéu
Carteira de couro
Brinco Argola

7. Nome do produto de maior preço de custo, desde que não tenha sido vendido ainda. Caso seja necessário mais de um SELECT, use operador de conjunto em algum deles.

nome character varying(40)
Carteira de couro