

/\* -------------------------------------------------------------------------- \*/

/\*Criação da tabela com seus determinados campos\*/

CREATE TABLE CLIENTE

(

CD\_CLIENTE INTEGER NOT NULL ,

NM\_CLIENTE VARCHAR(50) ,

NU\_CPF\_CNPJ INTEGER ,

TP\_CLIENTE VARCHAR(20) ,

CONSTRAINT CONS\_CLIENTE PRIMARY KEY (CD\_CLIENTE) ENABLE /\*chave primaria da tabela de Cliente\*/

);

/\* -------------------------------------------------------------------------- \*/

/\*Mostra todos os dados da tabela de Cliente\*/

SELECT \* FROM CLIENTE;

/\* -------------------------------------------------------------------------- \*/

/\*Inserindo dados na tabela de Cliente \*/

INSERT INTO CLIENTE( CD\_CLIENTE, NM\_CLIENTE, NU\_CPF\_CNPJ, TP\_CLIENTE )

VALUES ( 123,'Laís Ingrid',08967701411,'feirante' ) ;

INSERT INTO CLIENTE( CD\_CLIENTE, NM\_CLIENTE, NU\_CPF\_CNPJ, TP\_CLIENTE )

VALUES( 456,'Edna Soares',95687216852,'comprador' );

INSERT INTO CLIENTE( CD\_CLIENTE, NM\_CLIENTE, NU\_CPF\_CNPJ, TP\_CLIENTE )

VALUES( 789,'Iracy da Silva',65585284258,'exportador' );

COMMIT; /\*Comando que grava definitivamente os efeitos dos comandos de uma transação (insert, delete e update)\*/

/\* -------------------------------------------------------------------------- \*/

/\*Alteração de dados da Tabela de CLIENTE \*/

ALTER TABLE CLIENTE ADD (TESTE\_CLIENTE INTEGER);

ALTER TABLE CLIENTE MODIFY (TESTE\_CLIENTE VARCHAR(60));

ALTER TABLE CLIENTE DROP COLUMN TESTE\_CLIENTE;

/\* -------------------------------------------------------------------------- \*/

/\*Deletando o dados especifico da tabela de cliente quando o mesmo tiver á chave primaria = 123\*/

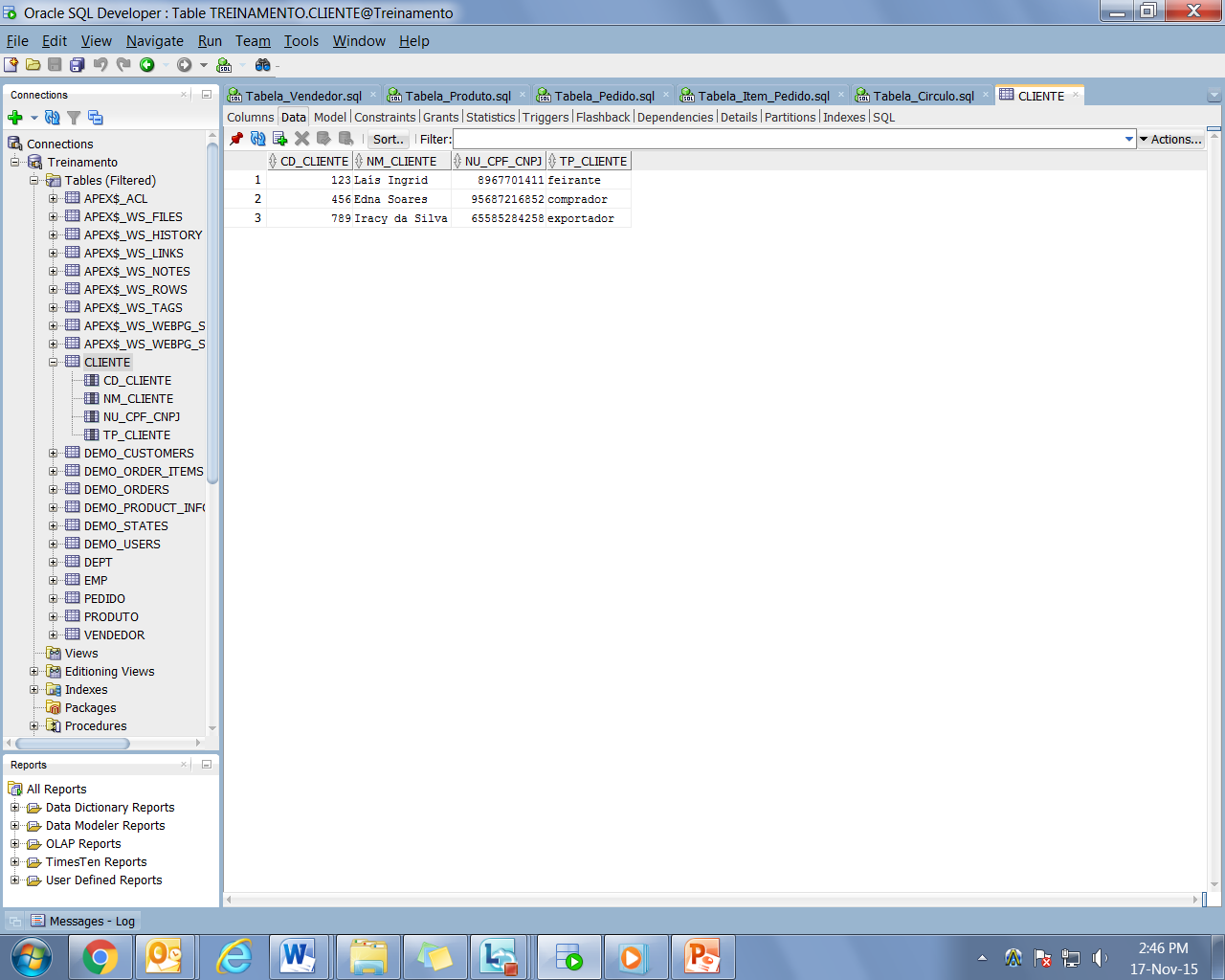
DELETE FROM CLIENTE WHERE CD\_CLIENTE = 123; /\*deleta á linha requerida na clausura where \*/

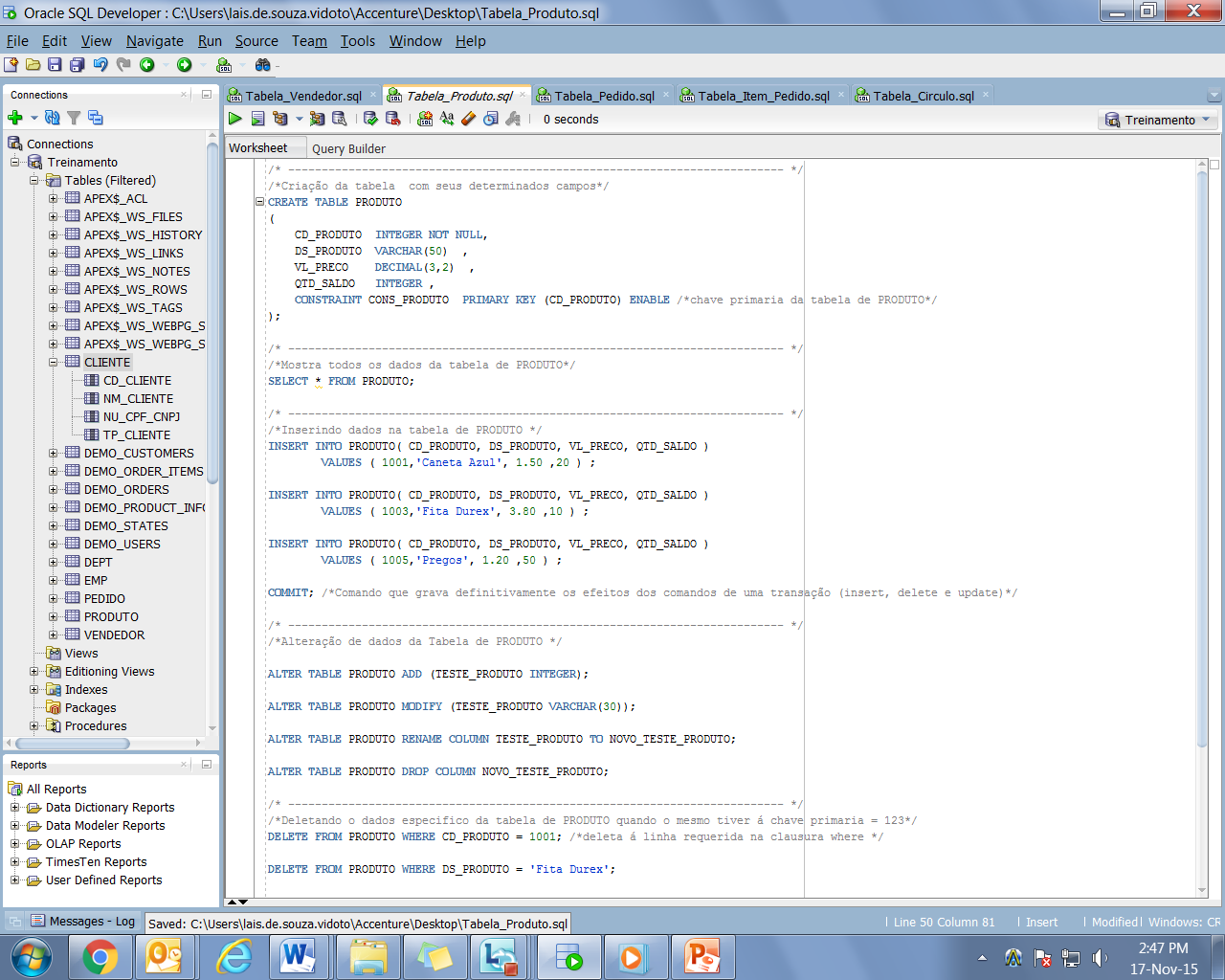
DELETE FROM CLIENTE WHERE NM\_CLIENTE = 'Edna Soares';

DELETE FROM CLIENTE WHERE NU\_CPF\_CNPJ = 65585284258;

COMMIT;

/\* -------------------------------------------------------------------------- \*/





/\* -------------------------------------------------------------------------- \*/

/\*Criação da tabela com seus determinados campos\*/

CREATE TABLE PRODUTO

(

CD\_PRODUTO INTEGER NOT NULL,

DS\_PRODUTO VARCHAR(50) ,

VL\_PRECO DECIMAL(3,2) ,

QTD\_SALDO INTEGER ,

CONSTRAINT CONS\_PRODUTO PRIMARY KEY (CD\_PRODUTO) ENABLE /\*chave primaria da tabela de PRODUTO\*/

);

/\* -------------------------------------------------------------------------- \*/

/\*Mostra todos os dados da tabela de PRODUTO\*/

SELECT \* FROM PRODUTO;

/\* -------------------------------------------------------------------------- \*/

/\*Inserindo dados na tabela de PRODUTO \*/

INSERT INTO PRODUTO( CD\_PRODUTO, DS\_PRODUTO, VL\_PRECO, QTD\_SALDO )

VALUES ( 1001,'Caneta Azul', 1.50 ,20 ) ;

INSERT INTO PRODUTO( CD\_PRODUTO, DS\_PRODUTO, VL\_PRECO, QTD\_SALDO )

VALUES ( 1003,'Fita Durex', 3.80 ,10 ) ;

INSERT INTO PRODUTO( CD\_PRODUTO, DS\_PRODUTO, VL\_PRECO, QTD\_SALDO )

VALUES ( 1005,'Pregos', 1.20 ,50 ) ;

COMMIT; /\*Comando que grava definitivamente os efeitos dos comandos de uma transação (insert, delete e update)\*/

/\* -------------------------------------------------------------------------- \*/

/\*Alteração de dados da Tabela de PRODUTO \*/

ALTER TABLE PRODUTO ADD (TESTE\_PRODUTO INTEGER);

ALTER TABLE PRODUTO MODIFY (TESTE\_PRODUTO VARCHAR(30));

ALTER TABLE PRODUTO RENAME COLUMN TESTE\_PRODUTO TO NOVO\_TESTE\_PRODUTO;

ALTER TABLE PRODUTO DROP COLUMN NOVO\_TESTE\_PRODUTO;

/\* -------------------------------------------------------------------------- \*/

/\*Deletando o dados especifico da tabela de PRODUTO quando o mesmo tiver á chave primaria = 123\*/

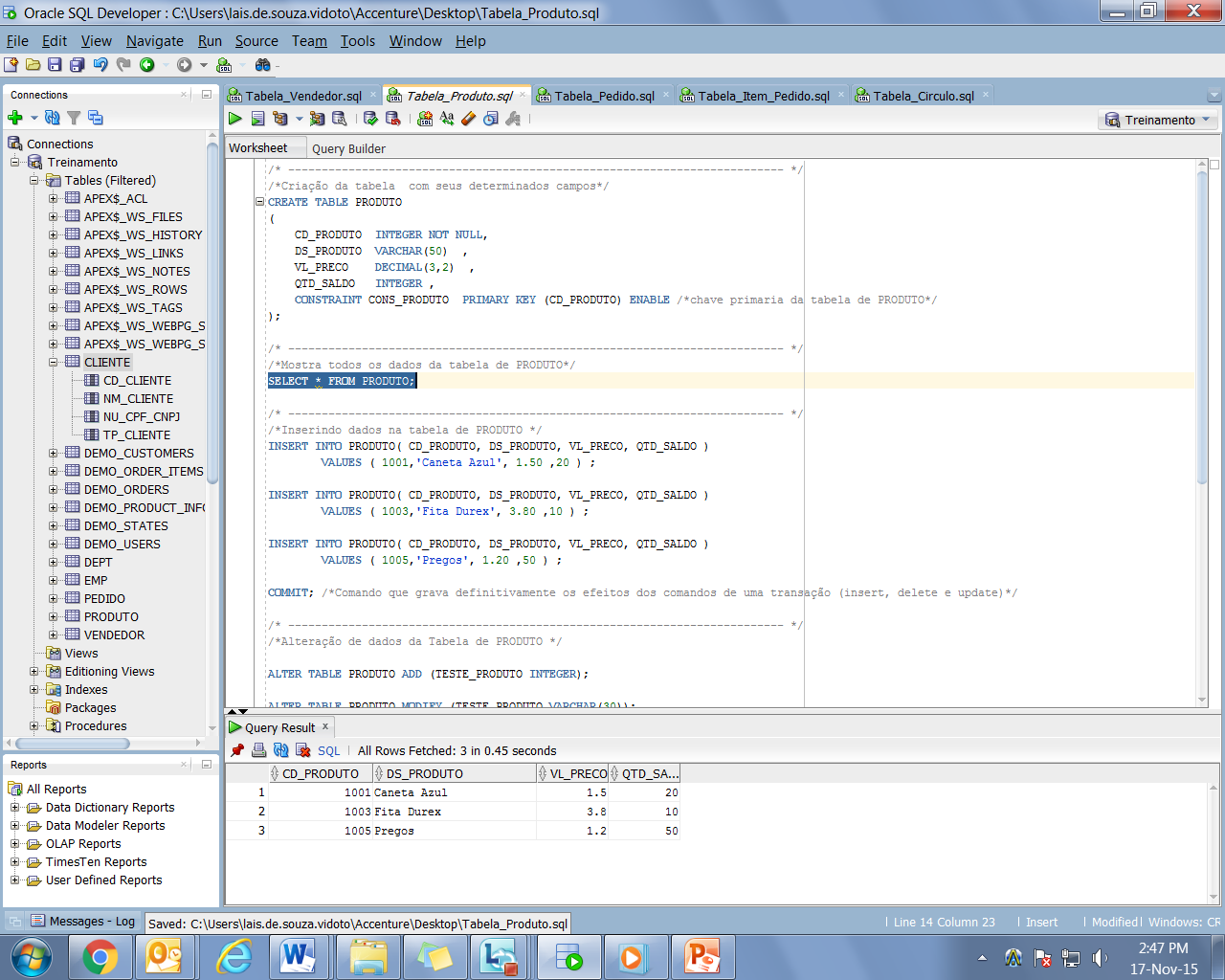
DELETE FROM PRODUTO WHERE CD\_PRODUTO = 1001; /\*deleta á linha requerida na clausura where \*/

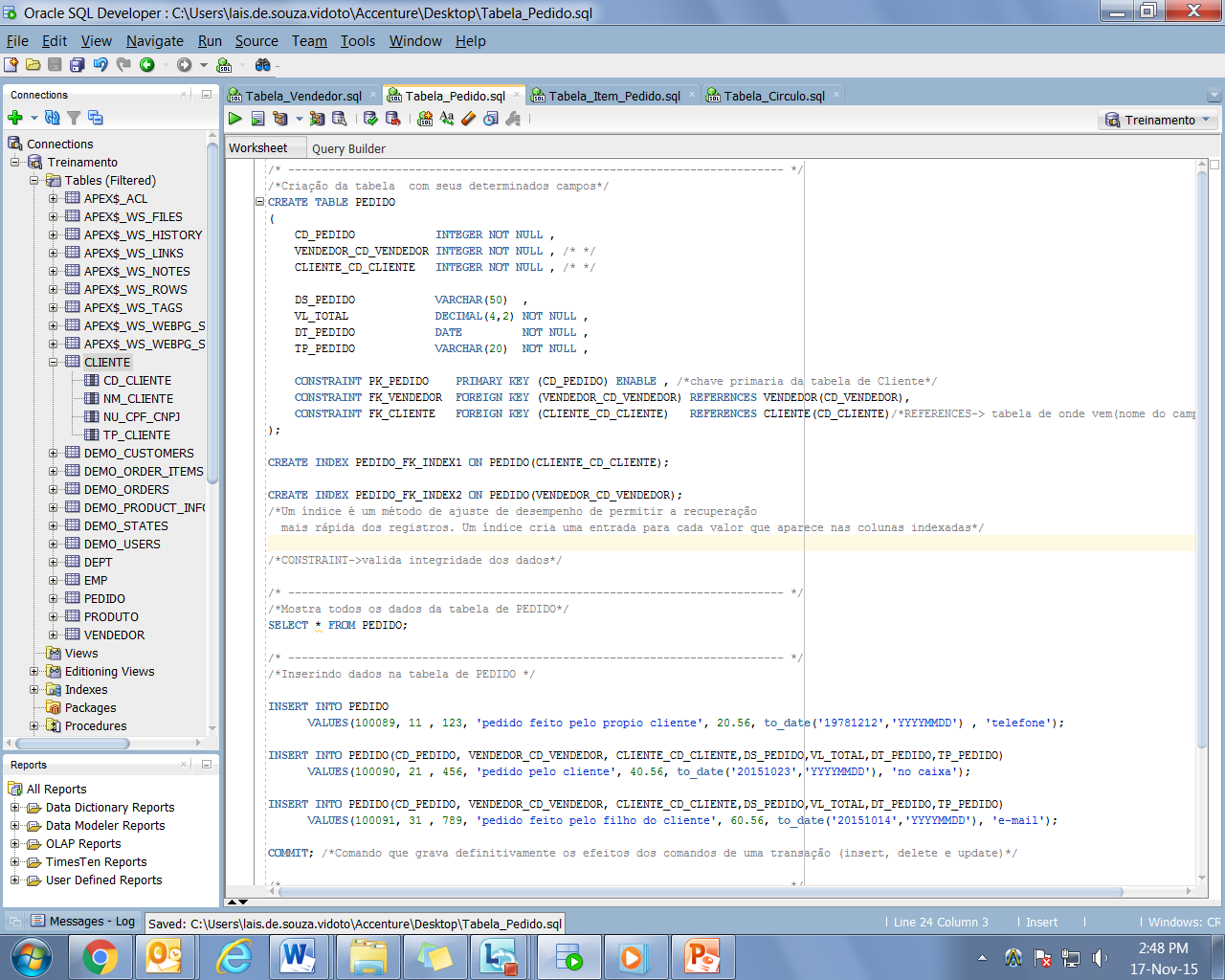
DELETE FROM PRODUTO WHERE DS\_PRODUTO = 'Fita Durex';

DELETE FROM PRODUTO WHERE VL\_PRECO = 1.20;

COMMIT;

/\* -------------------------------------------------------------------------- \*/





/\* -------------------------------------------------------------------------- \*/

/\*Criação da tabela com seus determinados campos\*/

CREATE TABLE PEDIDO

(

CD\_PEDIDO INTEGER NOT NULL ,

VENDEDOR\_CD\_VENDEDOR INTEGER NOT NULL , /\* \*/

CLIENTE\_CD\_CLIENTE INTEGER NOT NULL , /\* \*/

DS\_PEDIDO VARCHAR(50) ,

VL\_TOTAL DECIMAL(4,2) NOT NULL ,

DT\_PEDIDO DATE NOT NULL ,

TP\_PEDIDO VARCHAR(20) NOT NULL ,

CONSTRAINT PK\_PEDIDO PRIMARY KEY (CD\_PEDIDO) ENABLE , /\*chave primaria da tabela de Cliente\*/

CONSTRAINT FK\_VENDEDOR FOREIGN KEY (VENDEDOR\_CD\_VENDEDOR) REFERENCES VENDEDOR(CD\_VENDEDOR),

CONSTRAINT FK\_CLIENTE FOREIGN KEY (CLIENTE\_CD\_CLIENTE) REFERENCES CLIENTE(CD\_CLIENTE)/\*REFERENCES-> tabela de onde vem(nome do campo dela)\*/

);

CREATE INDEX PEDIDO\_FK\_INDEX1 ON PEDIDO(CLIENTE\_CD\_CLIENTE);

CREATE INDEX PEDIDO\_FK\_INDEX2 ON PEDIDO(VENDEDOR\_CD\_VENDEDOR);

/\*Um índice é um método de ajuste de desempenho de permitir a recuperação

mais rápida dos registros. Um índice cria uma entrada para cada valor que aparece nas colunas indexadas\*/

/\*CONSTRAINT->valida integridade dos dados\*/

/\* -------------------------------------------------------------------------- \*/

/\*Mostra todos os dados da tabela de PEDIDO\*/

SELECT \* FROM PEDIDO;

/\* -------------------------------------------------------------------------- \*/

/\*Inserindo dados na tabela de PEDIDO \*/

INSERT INTO PEDIDO

VALUES(100089, 11 , 123, 'pedido feito pelo propio cliente', 20.56, to\_date('19781212','YYYYMMDD') , 'telefone');

INSERT INTO PEDIDO(CD\_PEDIDO, VENDEDOR\_CD\_VENDEDOR, CLIENTE\_CD\_CLIENTE,DS\_PEDIDO,VL\_TOTAL,DT\_PEDIDO,TP\_PEDIDO)

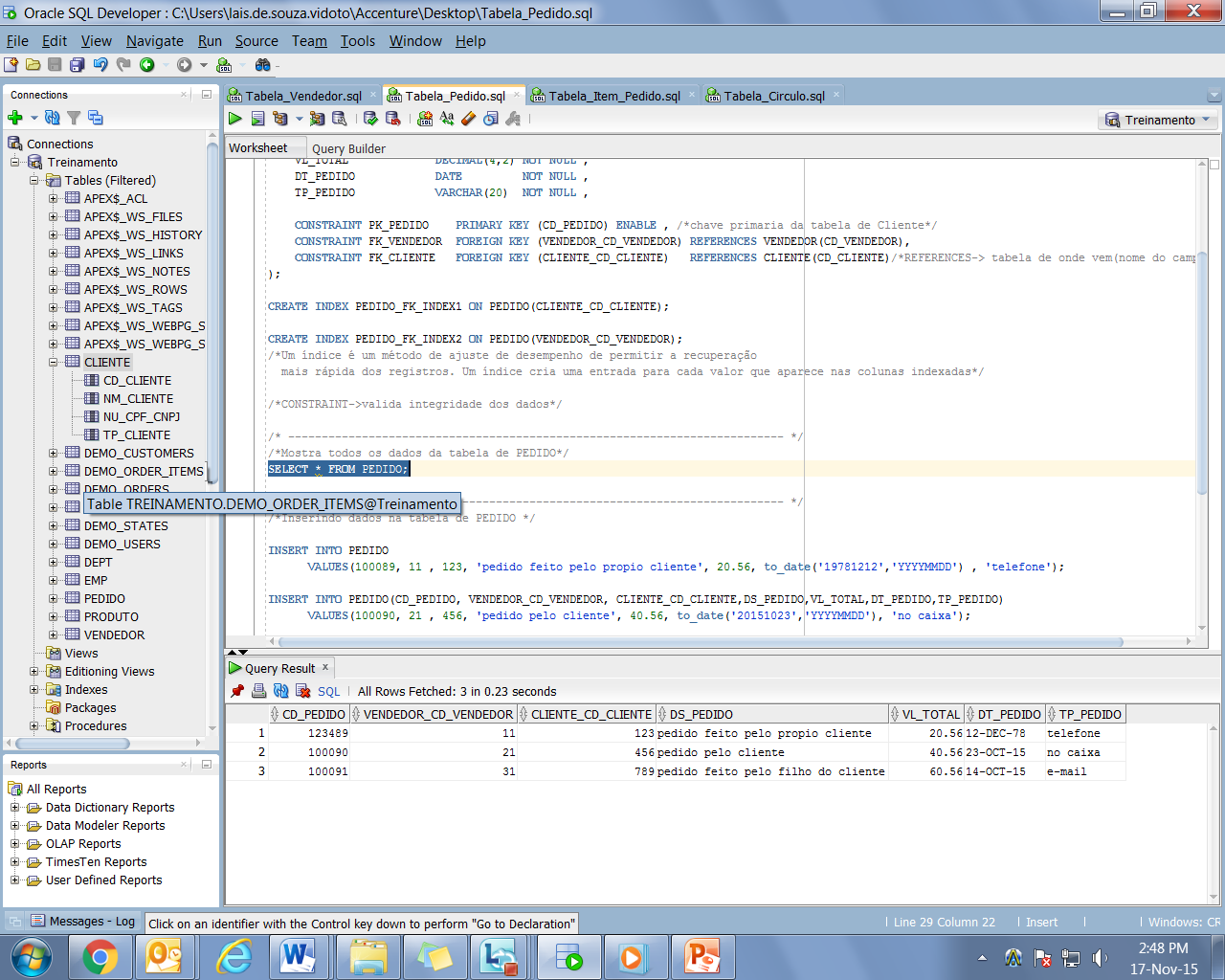
VALUES(100090, 21 , 456, 'pedido pelo cliente', 40.56, to\_date('20151023','YYYYMMDD'), 'no caixa');

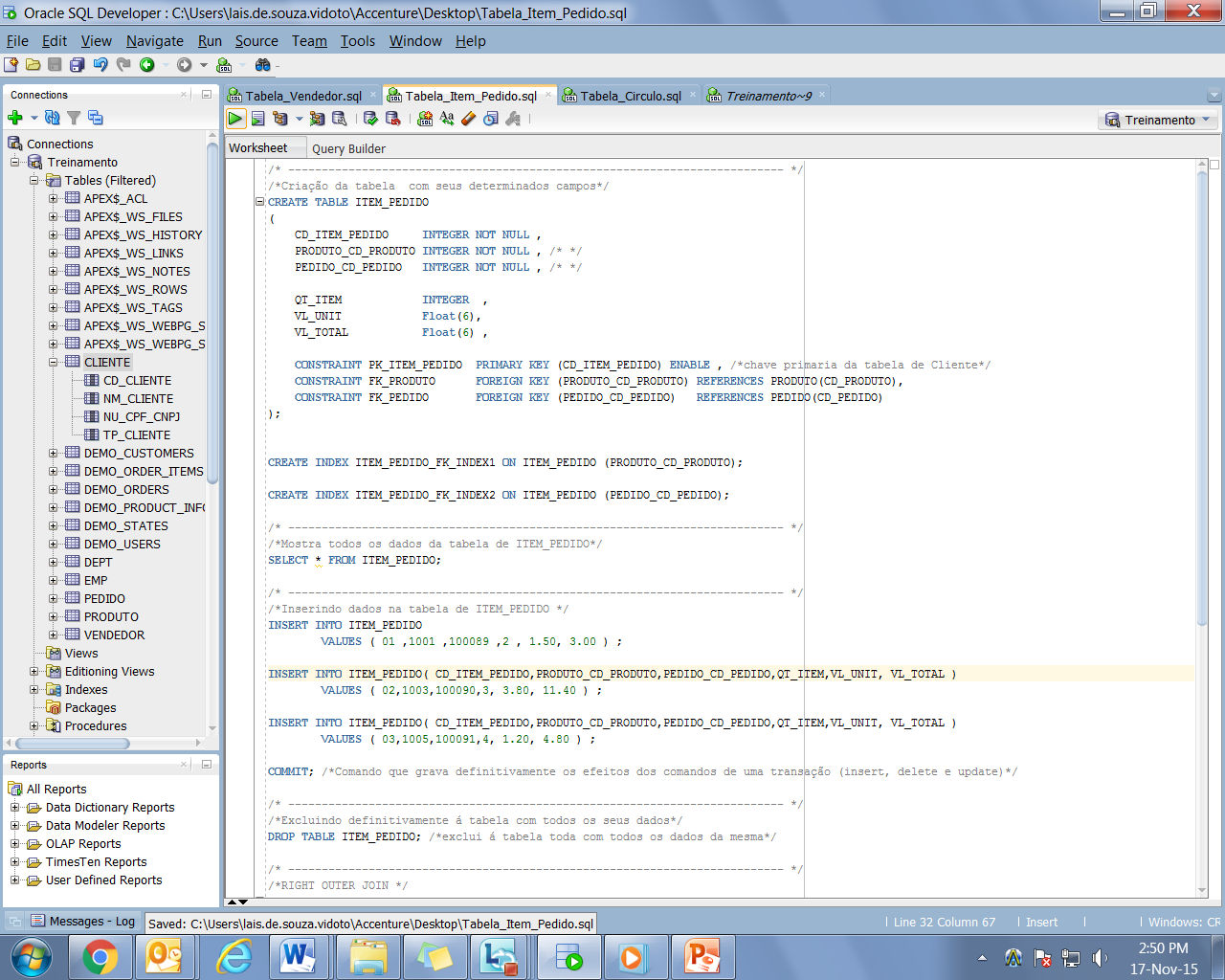
INSERT INTO PEDIDO(CD\_PEDIDO, VENDEDOR\_CD\_VENDEDOR, CLIENTE\_CD\_CLIENTE,DS\_PEDIDO,VL\_TOTAL,DT\_PEDIDO,TP\_PEDIDO)

VALUES(100091, 31 , 789, 'pedido feito pelo filho do cliente', 60.56, to\_date('20151014','YYYYMMDD'), 'e-mail');

COMMIT; /\*Comando que grava definitivamente os efeitos dos comandos de uma transação (insert, delete e update)\*/

/\* -------------------------------------------------------------------------- \*/





/\* -------------------------------------------------------------------------- \*/

/\*Criação da tabela com seus determinados campos\*/

CREATE TABLE ITEM\_PEDIDO

(

CD\_ITEM\_PEDIDO INTEGER NOT NULL ,

PRODUTO\_CD\_PRODUTO INTEGER NOT NULL , /\* \*/

PEDIDO\_CD\_PEDIDO INTEGER NOT NULL , /\* \*/

QT\_ITEM INTEGER ,

VL\_UNIT Float(6),

VL\_TOTAL Float(6) ,

CONSTRAINT PK\_ITEM\_PEDIDO PRIMARY KEY (CD\_ITEM\_PEDIDO) ENABLE , /\*chave primaria da tabela de Cliente\*/

CONSTRAINT FK\_PRODUTO FOREIGN KEY (PRODUTO\_CD\_PRODUTO) REFERENCES PRODUTO(CD\_PRODUTO),

CONSTRAINT FK\_PEDIDO FOREIGN KEY (PEDIDO\_CD\_PEDIDO) REFERENCES PEDIDO(CD\_PEDIDO)

);

CREATE INDEX ITEM\_PEDIDO\_FK\_INDEX1 ON ITEM\_PEDIDO (PRODUTO\_CD\_PRODUTO);

CREATE INDEX ITEM\_PEDIDO\_FK\_INDEX2 ON ITEM\_PEDIDO (PEDIDO\_CD\_PEDIDO);

/\* -------------------------------------------------------------------------- \*/

/\*Mostra todos os dados da tabela de ITEM\_PEDIDO\*/

SELECT \* FROM ITEM\_PEDIDO;

/\* -------------------------------------------------------------------------- \*/

/\*Inserindo dados na tabela de ITEM\_PEDIDO \*/

INSERT INTO ITEM\_PEDIDO

VALUES ( 01 ,1001 ,100089 ,2 , 1.50, 3.00 ) ;

INSERT INTO ITEM\_PEDIDO( CD\_ITEM\_PEDIDO,PRODUTO\_CD\_PRODUTO,PEDIDO\_CD\_PEDIDO,QT\_ITEM,VL\_UNIT, VL\_TOTAL )

VALUES ( 02,1003,100090,3, 3.80, 11.40 ) ;

INSERT INTO ITEM\_PEDIDO( CD\_ITEM\_PEDIDO,PRODUTO\_CD\_PRODUTO,PEDIDO\_CD\_PEDIDO,QT\_ITEM,VL\_UNIT, VL\_TOTAL )

VALUES ( 03,1005,100091,4, 1.20, 4.80 ) ;

COMMIT; /\*Comando que grava definitivamente os efeitos dos comandos de uma transação (insert, delete e update)\*/

/\* -------------------------------------------------------------------------- \*/

/\*Excluindo definitivamente á tabela com todos os seus dados\*/

DROP TABLE ITEM\_PEDIDO; /\*exclui á tabela toda com todos os dados da mesma\*/

/\* -------------------------------------------------------------------------- \*/

/\*RIGHT OUTER JOIN \*/

SELECT ITEM\_PEDIDO.CD\_ITEM\_PEDIDO ,

ITEM\_PEDIDO.PRODUTO\_CD\_PRODUTO ,

ITEM\_PEDIDO.PEDIDO\_CD\_PEDIDO,

ITEM\_PEDIDO.QT\_ITEM ,

ITEM\_PEDIDO.VL\_UNIT ,

ITEM\_PEDIDO.VL\_TOTAL

FROM ITEM\_PEDIDO

RIGHT OUTER JOIN PEDIDO

ON ITEM\_PEDIDO.PEDIDO\_CD\_PEDIDO = PEDIDO.CD\_PEDIDO

RIGHT OUTER JOIN PRODUTO

ON ITEM\_PEDIDO.PRODUTO\_CD\_PRODUTO = PRODUTO.CD\_PRODUTO;

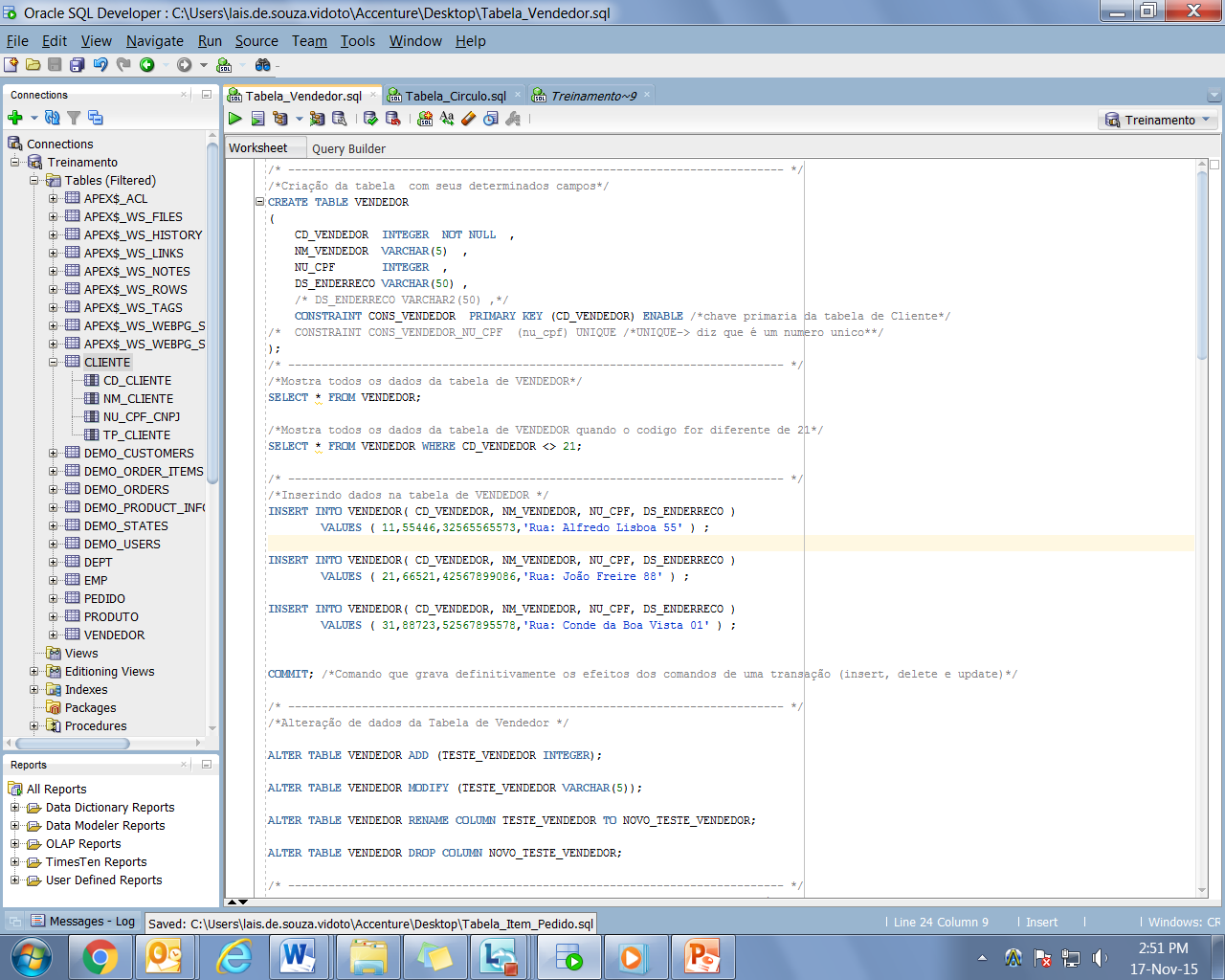
/\* -------------------------------------------------------------------------- \*/

SELECT PRODUTO\_CD\_PRODUTO FROM ITEM\_PEDIDO

UNION

SELECT CD\_PRODUTO FROM PRODUTO;

/\* -------------------------------------------------------------------------- \*/



/\* -------------------------------------------------------------------------- \*/

/\*Criação da tabela com seus determinados campos\*/

CREATE TABLE VENDEDOR

(

CD\_VENDEDOR INTEGER NOT NULL ,

NM\_VENDEDOR VARCHAR(5) ,

NU\_CPF INTEGER ,

DS\_ENDERRECO VARCHAR(50) ,

/\* DS\_ENDERRECO VARCHAR2(50) ,\*/

CONSTRAINT CONS\_VENDEDOR PRIMARY KEY (CD\_VENDEDOR) ENABLE /\*chave primaria da tabela de Cliente\*/

/\* CONSTRAINT CONS\_VENDEDOR\_NU\_CPF (nu\_cpf) UNIQUE /\*UNIQUE-> diz que é um numero unico\*\*/

);

/\* -------------------------------------------------------------------------- \*/

/\*Mostra todos os dados da tabela de VENDEDOR\*/

SELECT \* FROM VENDEDOR;

/\*Mostra todos os dados da tabela de VENDEDOR quando o codigo for diferente de 21\*/

SELECT \* FROM VENDEDOR WHERE CD\_VENDEDOR <> 21;

/\* -------------------------------------------------------------------------- \*/

/\*Inserindo dados na tabela de VENDEDOR \*/

INSERT INTO VENDEDOR( CD\_VENDEDOR, NM\_VENDEDOR, NU\_CPF, DS\_ENDERRECO )

VALUES ( 11,55446,32565565573,'Rua: Alfredo Lisboa 55' ) ;

INSERT INTO VENDEDOR( CD\_VENDEDOR, NM\_VENDEDOR, NU\_CPF, DS\_ENDERRECO )

VALUES ( 21,66521,42567899086,'Rua: João Freire 88' ) ;

INSERT INTO VENDEDOR( CD\_VENDEDOR, NM\_VENDEDOR, NU\_CPF, DS\_ENDERRECO )

VALUES ( 31,88723,52567895578,'Rua: Conde da Boa Vista 01' ) ;

COMMIT; /\*Comando que grava definitivamente os efeitos dos comandos de uma transação (insert, delete e update)\*/

/\* -------------------------------------------------------------------------- \*/

/\*Alteração de dados da Tabela de Vendedor \*/

ALTER TABLE VENDEDOR ADD (TESTE\_VENDEDOR INTEGER);

ALTER TABLE VENDEDOR MODIFY (TESTE\_VENDEDOR VARCHAR(5));

ALTER TABLE VENDEDOR RENAME COLUMN TESTE\_VENDEDOR TO NOVO\_TESTE\_VENDEDOR;

ALTER TABLE VENDEDOR DROP COLUMN NOVO\_TESTE\_VENDEDOR;

/\* -------------------------------------------------------------------------- \*/

/\*Deletando o dados especifico da tabela de VENDEDOR quando o mesmo tiver á chave primaria = 123\*/

DELETE FROM VENDEDOR WHERE CD\_VENDEDOR = 11; /\*deleta á linha requerida na clausura where \*/

DELETE FROM VENDEDOR WHERE NM\_VENDEDOR = 66521;

DELETE FROM VENDEDOR WHERE NU\_CPF = 52567895578;

COMMIT;

/\* -------------------------------------------------------------------------- \*/

/\*Atualiza os dados da tabela VENDEDOR, mudando o numero do vendedor para 12345

quando o codigo dele for > 10,

obs: ele vai atualizar apenas o campo do set com o valor descrito no codigo\*/

UPDATE VENDEDOR

SET NM\_VENDEDOR = 12345

WHERE CD\_VENDEDOR > 10;

/\* -------------------------------------------------------------------------- \*/

UPDATE VENDEDOR SET NM\_VENDEDOR = ( SELECT VENDEDOR\_CD\_VENDEDOR DS\_PEDIDO

FROM PEDIDO

WHERE PEDIDO.VENDEDOR\_CD\_VENDEDOR = VENDEDOR.CD\_VENDEDOR

)WHERE CD\_VENDEDOR > 10;

/\* -------------------------------------------------------------------------- \*/

/\*

mostra todas as colunas da tabela de vendedor quando á 6ª letra da descrição do enderreço

começar pela letra 'A'

\*/

SELECT \*

FROM VENDEDOR

WHERE DS\_ENDERRECO LIKE '\_\_\_\_\_A%';

/\*

mostra todas as colunas da tabela de vendedor quando á 6ª letra da descrição do enderreço

terminar pelo numero '8'

\*/

SELECT \*

FROM VENDEDOR

WHERE DS\_ENDERRECO LIKE '%8';

/\*

mostra todas as colunas da tabela de vendedor quando o codigo do vendedor for 11 e ou 21

estar com os valores .

\*/

SELECT \*

FROM VENDEDOR

WHERE CD\_VENDEDOR IN (11,21);

/\*

mostra todas as colunas da tabela de vendedor quando o codigo do vendedor estiver entre os valores abaixo.

\*/

SELECT \*

FROM vendedor

WHERE CD\_VENDEDOR BETWEEN 10 AND 30;

/\* -------------------------------------------------------------------------- \*/

