

Nome dos alunos integrantes da atividade:

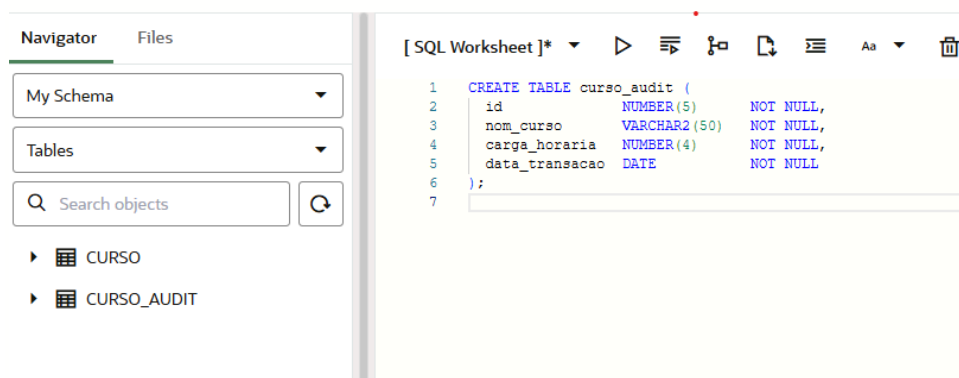
- Felipe Martins
- Lucas Tanaka
- Pedro Gabriel
- Yuri Costa

Trigger 1

A) Crie a tabela curso_audit para auditoria do curso

Utilizamos o comando:

```
CREATE TABLE curso_audit (  
  id          NUMBER(5)  NOT NULL,  
  nom_curso   VARCHAR2(50) NOT NULL,  
  carga_horaria NUMBER(4)  NOT NULL,  
  data_transacao SYSDATE    NOT NULL  
);
```



B) Crie a sequence seq_audit para utilizar no insert dos registros da curso_audit.

Utilizamos o comando:

```
CREATE SEQUENCE seq_audit  
  START WITH 1  
  INCREMENT BY 1  
  NOCYCLE;
```

```
INSERT INTO curso_audit (id, nom_curso, carga_horaria, data_transacao)  
VALUES (seq_audit.NEXTVAL, 'Lógica de Programação', 160, SYSDATE);
```

[SQL Worksheet]*

```

1  CREATE SEQUENCE seq_audit
2  START WITH 1
3  INCREMENT BY 1
4  NOCYCLE;
5
6
7  INSERT INTO curso_audit (id, nom_curso, carga_horaria, data_transacao)
8  VALUES (seq_audit.NEXTVAL, 'Biologia Microscópica', 300 , SYSDATE);
9
10 SELECT * FROM CURSO_AUDIT
11

```

Query result Script output DBMS output Explain Plan SQL history

Download Execution time: 0.002 seconds

	ID	NOM_CURSO	CARGA_HORARIA	DATA_TRANSACAO
1	1	Lógica de Programaç	160	5/20/2025, 10:55:5
2	3	Biologia Microscópica	300	5/20/2025, 11:09:58
3	2	Ciência da computaç	3000	5/20/2025, 11:09:2'

E o select para exibir em tela os valores inseridos.

C) Crie uma trigger na tabela CURSO para que o registro excluído com carga horária >= 150, seja registrado na tabela CURSO_AUDIT. Em caso de inserção de novo curso, atribuir zero no campo carga horária anterior.

Utilizamos o comando:

```

CREATE OR REPLACE TRIGGER trg_curso_audit
BEFORE INSERT OR DELETE ON curso
FOR EACH ROW
DECLARE
    v_id NUMBER;
BEGIN
    -- Inserção: força carga_horaria_ant = 0
    IF INSERTING THEN
        :NEW.carga_horaria_ant := 0;

    -- Exclusão: se carga_horaria >= 150, registra na curso_audit
    ELSIF DELETING THEN
        IF :OLD.carga_horaria >= 150 THEN

```

```

SELECT seq_audit.NEXTVAL INTO v_id FROM dual;
INSERT INTO curso_audit (id, nom_curso, carga_horaria, data_transacao)
VALUES (v_id, :OLD.nom_curso, :OLD.carga_horaria, SYSDATE);
END IF;
END IF;
END;

```

Utilizamos old pois é um valor antigo que está sendo alterado

```

[ SQL Worksheet ]*
1 CREATE OR REPLACE TRIGGER trg_curso_audit
2 BEFORE INSERT OR DELETE ON curso
3 FOR EACH ROW
4 DECLARE
5     v_id NUMBER;
6 BEGIN
7
8     IF INSERTING THEN
9         :NEW.carga_horaria_ant := 0;
10
11
12     ELSIF DELETING THEN
13         IF :OLD.carga_horaria >= 150 THEN
14             SELECT seq_audit.NEXTVAL INTO v_id FROM dual;
15             INSERT INTO curso_audit (id, nom_curso, carga_horaria, data_transacao)
16             VALUES (v_id, :OLD.nom_curso, :OLD.carga_horaria, SYSDATE);
17             END IF;
18         END IF;
19     END;
20

```

Query result | **Script output** | DBMS output | Explain Plan | SQL history

FOR EACH ROW
DECLARE...
Show more...

Trigger TRG_CURSO_AUDIT compiled
Elapsed: 00:00:00.019

D) Faça os testes da trigger.

Realizamos o teste de insert para inserir o novo valor.

```

[ SQL Worksheet ]*
1 INSERT INTO curso (id_curso, nom_curso, carga_horaria)
2 VALUES (1, 'Redes de Computadores', 160);
3
4

```

Query result | **Script output** | DBMS output | Explain Plan | SQL history

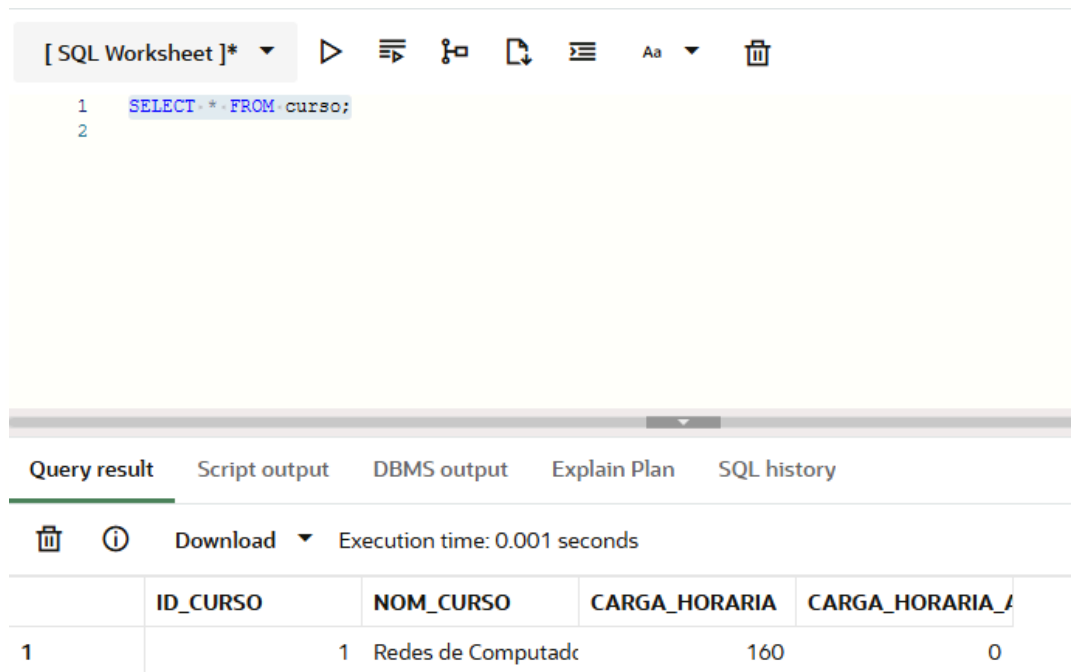
FOR EACH ROW
DECLARE...
Show more...

Trigger TRG_CURSO_AUDIT compiled
Elapsed: 00:00:00.019

SQL> INSERT INTO curso (id_curso, nom_curso, carga_horaria)
VALUES (1, 'Redes de Computadores', 160)

1 row inserted.
Elapsed: 00:00:00.223

O comando select foi utilizado para validar se a mudança realmente foi realizada



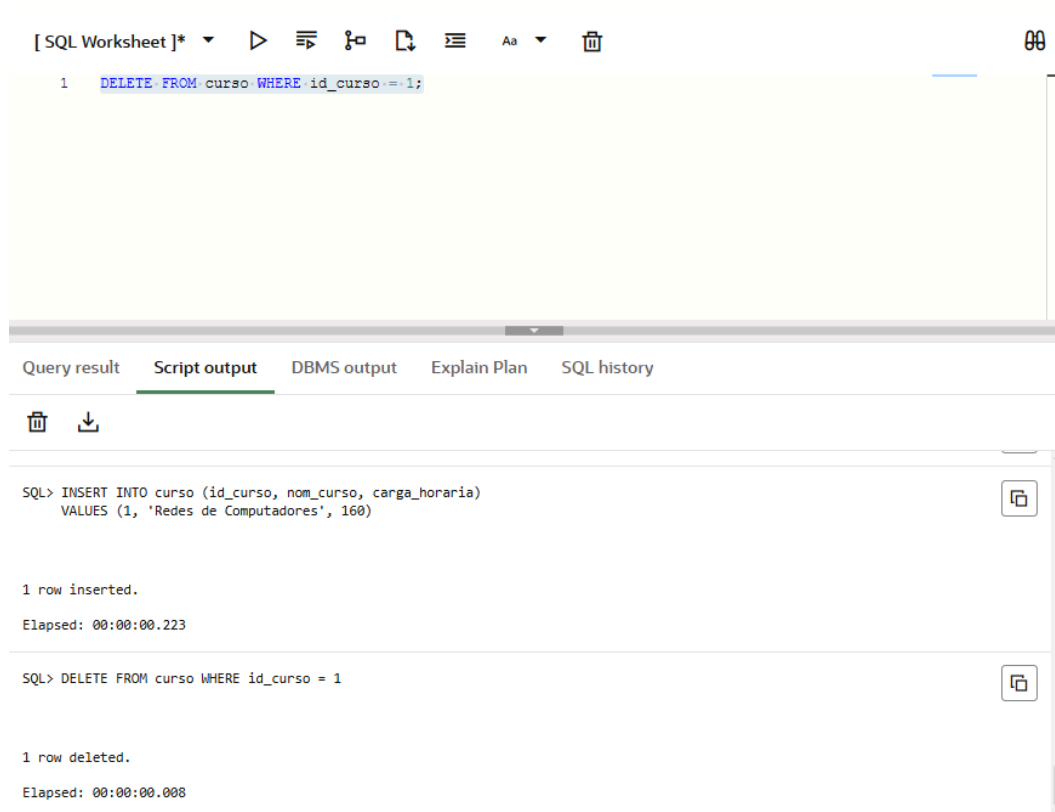
The screenshot shows a SQL Worksheet interface. At the top, there is a toolbar with icons for running queries, saving, and other functions. Below the toolbar, the SQL code is entered in a text area:

```
1 SELECT *. FROM curso;  
2
```

Below the code area, there are tabs for "Query result", "Script output", "DBMS output", "Explain Plan", and "SQL history". The "Query result" tab is selected, showing the results of the query. Above the table, there are icons for deleting and downloading, and a "Download" button. The execution time is displayed as "Execution time: 0.001 seconds".

	ID_CURSO	NOM_CURSO	CARGA_HORARIA	CARGA_HORARIA_A
1	1	Redes de Computad	160	0

O comando delete foi utilizado para apagar a informação de ID=1 da tabela



The screenshot shows a SQL Worksheet interface. At the top, there is a toolbar with icons for running queries, saving, and other functions. Below the toolbar, the SQL code is entered in a text area:

```
1 DELETE FROM curso WHERE id_curso = 1;
```

Below the code area, there are tabs for "Query result", "Script output", "DBMS output", "Explain Plan", and "SQL history". The "Script output" tab is selected, showing the output of the script. The output includes the following text:

```
SQL> INSERT INTO curso (id_curso, nom_curso, carga_horaria)  
VALUES (1, 'Redes de Computadores', 160)  
  
1 row inserted.  
Elapsed: 00:00:00.223  
  
SQL> DELETE FROM curso WHERE id_curso = 1  
  
1 row deleted.  
Elapsed: 00:00:00.008
```

Na imagem podemos perceber que a tabela da auditoria foi acrescentada com as informações que foram excluídas

[SQL Worksheet]*

Aa

1

SELECT * FROM curso_audit;

Query result

Script output

DBMS output

Explain Plan

SQL history

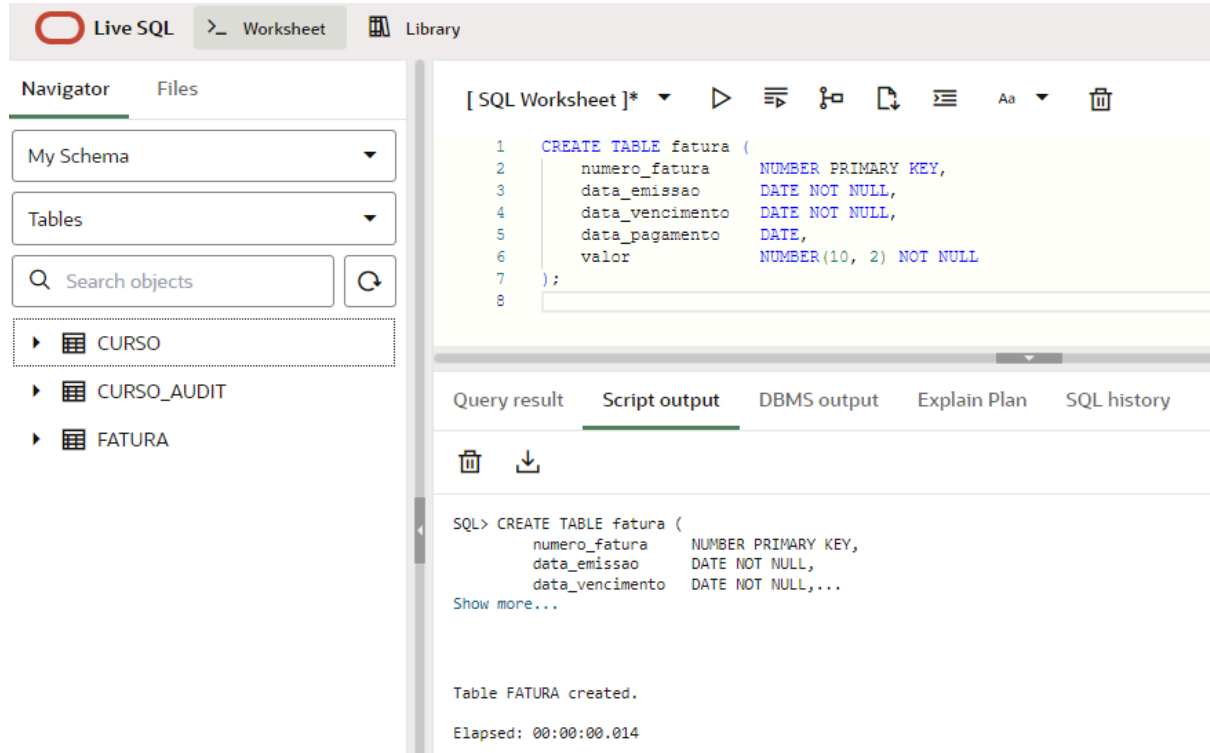
Download

Execution time: 0.001 seconds

	ID	NOM_CURSO	CARGA_HORARIA	DATA_TRANSACAO
1	1	Lógica de Programaç	160	5/20/2025, 10:55:56
2	3	Biologia Microscópica	300	5/20/2025, 11:09:56
3	2	Ciência da computaç	3000	5/20/2025, 11:09:26
4	4	Redes de Computad	160	5/20/2025, 11:22:26

TRIGGER 2

A) Crie a tabela Fatura com os campos: numero da fatura, data da emissão, data do vencimento, data do pagamento e valor. Defina uma chave primária.



The screenshot shows the Live SQL interface. On the left, the Navigator panel displays 'My Schema' and 'Tables'. The 'Tables' section lists 'CURSO', 'CURSO_AUDIT', and 'FATURA'. The main editor shows the SQL script to create the 'fatura' table:

```
1 CREATE TABLE fatura (  
2     numero_fatura    NUMBER PRIMARY KEY,  
3     data_emissao      DATE NOT NULL,  
4     data_vencimento   DATE NOT NULL,  
5     data_pagamento    DATE,  
6     valor             NUMBER(10, 2) NOT NULL  
7 );  
8
```

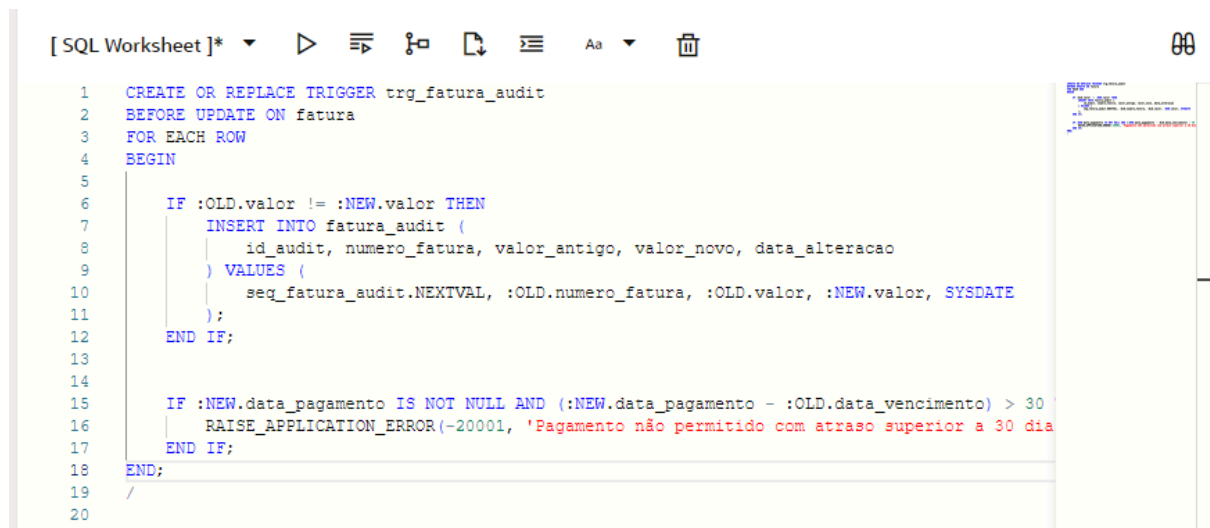
Below the editor, the 'Script output' tab shows the execution results:

```
SQL> CREATE TABLE fatura (  
      numero_fatura    NUMBER PRIMARY KEY,  
      data_emissao      DATE NOT NULL,  
      data_vencimento   DATE NOT NULL,...  
Show more...
```

Table FATURA created.
Elapsed: 00:00:00.014

Tabela criada e chave primária definida como o número da fatura.

B) Alterações realizadas no valor da fatura devem ser registradas em uma tabela de auditoria que contenha campos para rastreabilidade da informação.



The screenshot shows the Live SQL interface with the SQL script to create a trigger for auditing changes to the 'fatura' table:

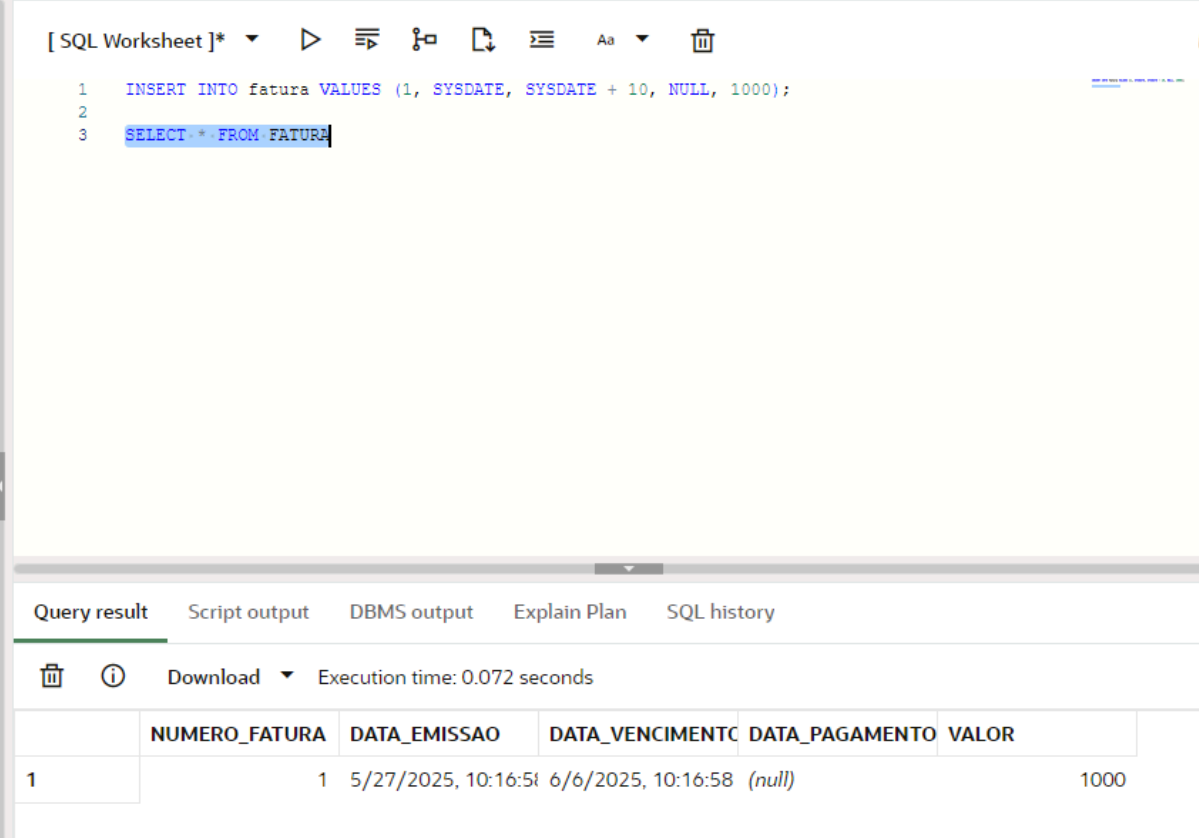
```
1 CREATE OR REPLACE TRIGGER trg_fatura_audit  
2 BEFORE UPDATE ON fatura  
3 FOR EACH ROW  
4 BEGIN  
5  
6     IF :OLD.valor != :NEW.valor THEN  
7         INSERT INTO fatura_audit (  
8             id_audit, numero_fatura, valor_antigo, valor_novo, data_alteracao  
9         ) VALUES (  
10            seq_fatura_audit.NEXTVAL, :OLD.numero_fatura, :OLD.valor, :NEW.valor, SYSDATE  
11        );  
12     END IF;  
13  
14  
15     IF :NEW.data_pagamento IS NOT NULL AND (:NEW.data_pagamento - :OLD.data_vencimento) > 30  
16     RAISE_APPLICATION_ERROR(-20001, 'Pagamento não permitido com atraso superior a 30 dias');  
17     END IF;  
18 END;  
19 /  
20
```

The right side of the interface shows a preview of the 'fatura_audit' table structure:

```
CREATE TABLE fatura_audit (  
  id_audit NUMBER(10, 0) NOT NULL,  
  numero_fatura NUMBER(10, 0) NOT NULL,  
  valor_antigo NUMBER(10, 2) NOT NULL,  
  valor_novo NUMBER(10, 2) NOT NULL,  
  data_alteracao DATE NOT NULL,  
  CONSTRAINT pk_fatura_audit PRIMARY KEY (id_audit)  
);
```

Criação da trigger para os pagamentos em atraso.

d) Faça os testes da trigger.



The screenshot shows a SQL IDE interface. The top panel is a SQL worksheet titled "[SQL Worksheet]*" containing the following SQL code:

```
1 INSERT INTO fatura VALUES (1, SYSDATE, SYSDATE + 10, NULL, 1000);
2
3 SELECT * FROM FATURA
```

The bottom panel displays the "Query result" tab, showing the execution time as 0.072 seconds. Below this, a table with 6 columns is shown:

	NUMERO_FATURA	DATA_EMISSAO	DATA_VENCIMENTO	DATA_PAGAMENTO	VALOR
1	1	5/27/2025, 10:16:58	6/6/2025, 10:16:58	(null)	1000

Insert e select na tabela fatura

[SQL Worksheet]*

Aa

1

2

3

4

UPDATE fatura SET valor = 1200 WHERE numero_fatura = 1;

SELECT * FROM FATURA_AUDIT

Query result

Script output

DBMS output

Explain Plan

SQL history

Download

Execution time: 0.007 seconds

	ID_AUDIT	NUMERO_FATURA	VALOR_ANTIGO	VALOR_NOVO	DATA_ALTERACAO
1	1	1	1000	1200	5/27/2025, 10:18:34

Update nos dados inseridos no insert e a visualização da tabela da auditoria.


```
[ SQL Worksheet ]* 1 UPDATE fatura SET data_pagamento = data_vencimento + 31 WHERE numero_fatura = 1;
```

Query result Script output DBMS output Explain Plan SQL history

SQL> INSERT INTO fatura VALUES (1, SYSDATE, SYSDATE + 10, NULL, 1000)

1 row inserted.

Elapsed: 00:00:00.021

SQL> UPDATE fatura SET valor = 1200 WHERE numero_fatura = 1

1 row updated.

Elapsed: 00:00:00.025

SQL> UPDATE fatura SET data_pagamento = data_vencimento + 31 WHERE numero_fatura = 1

ORA-20001: Pagamento não permitido com atraso superior a 30 dias.
ORA-06512: at "SQL_PD840KRNK8P75TMVIE410Y57HH.TRG_FATURA_AUDIT", line 13
ORA-04088: error during execution of trigger 'SQL_PD840KRNK8P75TMVIE410Y57HH.TRG_FATURA_AUDIT'

<https://docs.oracle.com/error-help/db/ora-20001/>
Error at Line: 4 Column: 0

Tentativa de lançar uma fatura com a data de pagamento superior a 30 dias.