

# Banco de Dados Ciclos de Estudos

## Ciclos de Estudo

—

Por Antonio Welton Ribeiro Souza

Para que se analise a eficiência do aprendizado é fundamental manter informações sobre o tempo despendido nos estudos.

# O que se registra no banco de dados Ciclo de Estudo?

Ciclo de Estudo é um banco de dados simples com a finalidade principal de permitir ao estudante registrar o horário que começou e que parou estudar, além disso permite analisar o assunto estudado bem como a disciplina, curso, área do conhecimento e o professor. Todos esses critérios serão associados permitindo uma visão ampla de cada ciclo de estudo.

---

# Ferramentas Utilizadas

Todas são gratuitas!

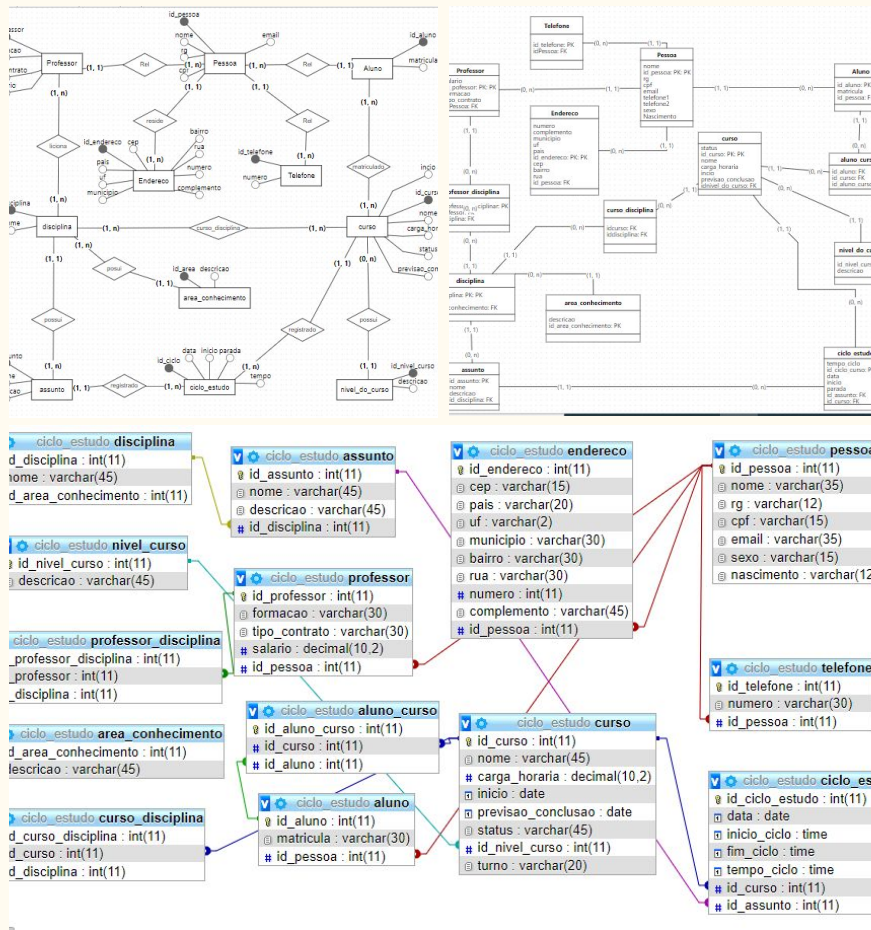
- br modelo-web
- XAMPP
- MYSQL
- MYSQL WorkBench



## 1. Modelagem Conceitual

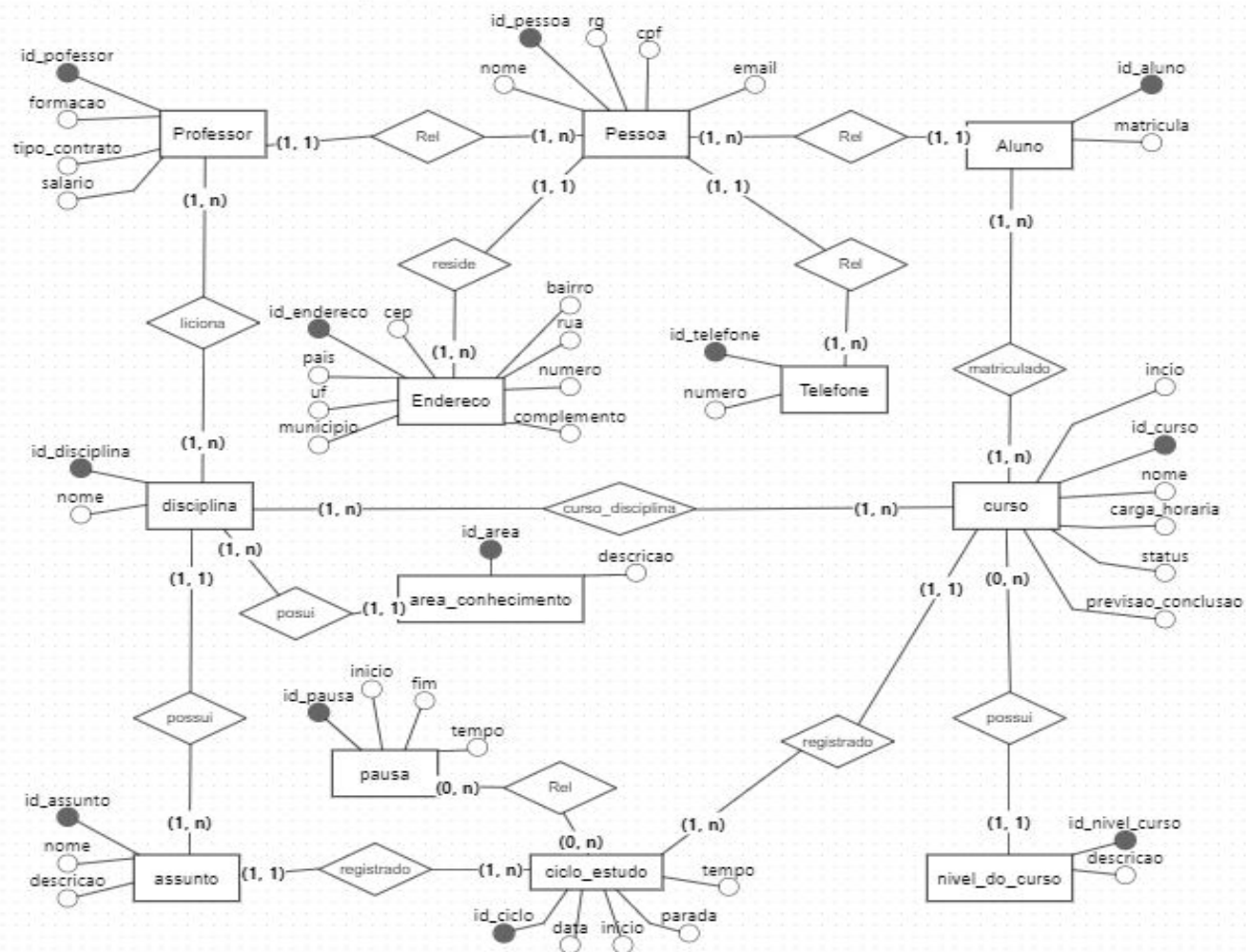
## 2. Modelagem Lógica

### 3. Modelagem Física



# Modelagem Conceitual

Na modelagem conceitual observa-se o Modelo Entidade Relacionamento(MER). Cria-se um diagrama com a finalidade de demonstrar as entidades, os atributos e os graus de relacionamento(1 para 1, 1,1; 1 para muitos, 1,m; muitos para muitos, m,m).

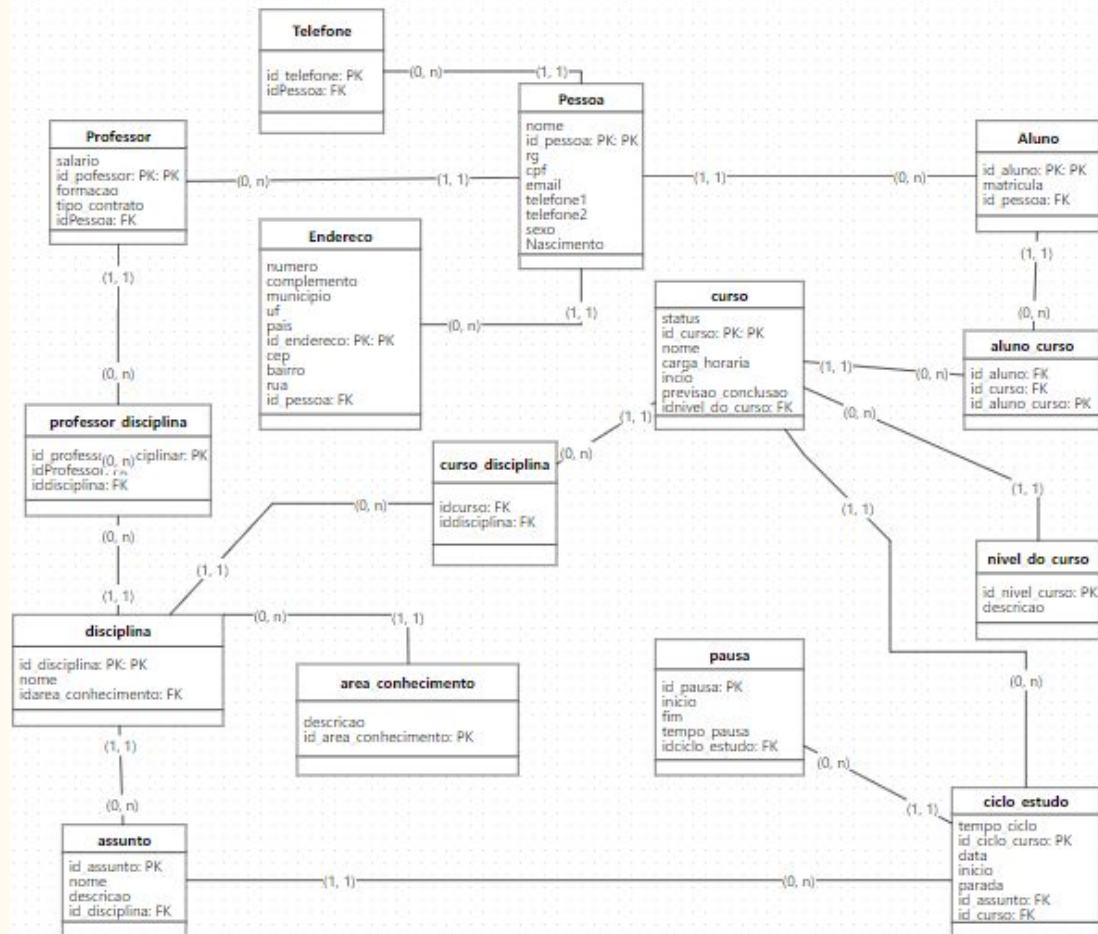


# Modelagem Lógica.

Agora se traduz as entidades, seus atributos e relacionamentos em tabelas compostas por linhas e colunas com a indicação de seus atributos chaves. Pode-se representar essa etapa por meio do diagrama lógico:

---





# Modelagem Física

A modelagem física é uma etapa que consiste na tradução do modelo lógico, ela serve para descrever as colunas, seus tipos de dados e as relações entre as tabelas. Percebe-se essa modelagem é, na verdade, a implementação do banco de dados por meio da linguagem SQL.

---

<b>ciclo_estudo telefone</b>
id_telefone : int(11)
numero : varchar(30)
# id_pessoa : int(11)

<b>ciclo_estudo disciplina</b>
id_disciplina : int(11)
nome : varchar(45)
# id_area_conhecimento : int(11)

<b>ciclo_estudo pessoa</b>
id_pessoa : int(11)
nome : varchar(35)
rg : varchar(12)
cpf : varchar(15)
email : varchar(35)
sexo : varchar(15)
nascimento : varchar(12)

<b>ciclo_estudo professor</b>
id_professor : int(11)
formacao : varchar(30)
tipo_contrato : varchar(30)
# salario : decimal(10,2)
# id_pessoa : int(11)

<b>ciclo_estudo aluno</b>
id_aluno : int(11)
matricula : varchar(30)
# id_pessoa : int(11)

<b>ciclo_estudo nivel_curso</b>
id_nivel_curso : int(11)
descricao : varchar(45)

<b>ciclo_estudo endereco</b>
id_endereco : int(11)
cep : varchar(15)
pais : varchar(20)
uf : varchar(2)
municipio : varchar(30)
bairro : varchar(30)
rua : varchar(30)
# numero : int(11)
# complemento : varchar(45)
# id_pessoa : int(11)

<b>ciclo_estudo assunto</b>
id_assunto : int(11)
nome : varchar(45)
descricao : varchar(45)
# id_disciplina : int(11)

<b>ciclo_estudo professor_disciplina</b>
id_professor_disciplina : int(11)
# id_professor : int(11)
# id_disciplina : int(11)

<b>ciclo_estudo aluno_curso</b>
id_aluno_curso : int(11)
# id_curso : int(11)
# id_aluno : int(11)

<b>ciclo_estudo curso</b>
id_curso : int(11)
nome : varchar(45)
# carga_horaria : decimal(10,2)
inicio : date
previsao_conclusao : date
status : varchar(45)
# id_nivel_curso : int(11)
turno : varchar(20)

<b>ciclo_estudo area_conhecimento</b>
id_area_conhecimento : int(11)
descricao : varchar(45)

<b>ciclo_estudo ciclo_estudo</b>
id_ciclo_estudo : int(11)
data : date
inicio_ciclo : time
fim_ciclo : time
tempo_ciclo : time
# id_curso : int(11)
# id_assunto : int(11)
tempo_liquido : time

<b>ciclo_estudo pausa</b>
id_pausa : int(11)
inicio_pausa : time
fim_pausa : time
tempo_pausa : time
# id_ciclo_estudo : int(11)

<b>ciclo_estudo curso_disciplina</b>
id_curso_disciplina : int(11)
# id_curso : int(11)
# id_disciplina : int(11)

# Inserts

São comandos que permitem que seja inserido dados nas tabelas.



```
-- DML
-- Area de Conhecimento
INSERT INTO `area_conhecimento` VALUES (8,'Humanas'),
(9,'Exatas'),
(10,'Biológicas'),
(11,'Químicas'),
(12,'Físicas'),
(13,'espaciais'),
(14,'Tecnologia da informação'),
(15,'Comunicação'),
(16,'Designer'),
(17,'Beleza e estética');

-- Nivel Curso
INSERT INTO `nivel_curso` VALUES (1,'Fundamental'),
(2,'Medio'),
(3,'Superior'),
(4,'Especialização'),
(5,'Mestrado'),
(6,'Doutorado');
```

-- Disciplina

```
INSERT INTO `disciplina` VALUES
(1,'Língua Portuguesa',15),
(2,'Tecnologia da Informação',14),
(3,'Microbiologia',10),
(4,'Raciocínio Lógico',9),
(5,'Matemática Básica',9),
(6,'Matemática Avançada',9),
(7,'Direito Constitucional',8);
```

-- Assunto

```
insert into assunto(nome, descricao, id_disciplina) values
('Lógica de primeira ordem!', 'quantificadores',4),
('Direitos e deveres individuais e coletivos', 'Direitos Políticos',7),
('Princípios Fundamentais da Contagem', 'Arranjos',5);
```

```
insert into assunto(nome, descricao, id_disciplina) values ('Programação Java', 'Java Collections',2);
```

-- Curso

```
insert into curso(nome, carga_horaria, inicio, previsao_conclusao, status, id_nivel_curso, turno) values
('Desenvolvimento-Web', 250, '2024-01-11', '2024-05-11', 'Ativo', 3, 'livre');
```

```
insert into curso(nome, carga_horaria, inicio, previsao_conclusao, status, id_nivel_curso, turno) values
('Relações Humanas', 250, '2024-02-11', '2024-06-11', 'Ativo', 3, 'tarde');
```



-- Pessoa

```
insert into pessoa(nome, rg, cpf, email, sexo, nascimento) values
('Marcos Franco','5552330-DF','22533366689','francomarcos@hotmail.com','masculino','1995-02-23'),
('Thaís Leal','2252330-MG','38533366689','tataleal@hotmail.com','feminino','1999-03-23'),
('Rafael Dourado','7766330-MG','77633366689','rafaeldourado@gmail.com','masculino','2005-03-23'),
('Roberto Santiago','20272330-BA','28533366614','robero101@hotmail.com','masculino','2002-03-30');
```

-- Professor

```
insert into professor(formacao, tipo_contrato, salario, id_pessoa) values
('Cientista de Dados', 'CLT-Indeterminado', 15000, 1),
('Letras Inglês e Português', 'CLT-Indeterminado', 9000, 2);
```

```
insert into endereco(cep, pais, uf, municipio, bairro, rua, numero, complemento, id_pessoa) values
('712530', 'Brasil', 'DF', 'Brasília', 'Asa Sul', '12', 133, 'casa',1),
('125341', 'Brasil', 'BA', 'Brasília', 'Asa Norte', '15', 233, 'casa',2),
('812532', 'Brasil', 'MG', 'Belo Horizonte', 'Centro', '12', 133, 'casa',3),
('412535', 'Brasil', 'RJ', 'Rio de Janeiro', 'Copacabana', '12', 133, 'casa',4);
```

-- Aluno

```
insert into aluno(matricula, id_pessoa) values ('GC-20245830', 3), ('GC-20245831', 4);
```

```
insert into telefone(numero, id_pessoa) values ('6199999999', 1), ('6199999988', 2), ('6199999977', 3), ('6199999966', 4);
```

# Tabelas associativas

As tabelas associativas surgem a partir do relacionamento de cardinalidade, muito para muitos (m,m), já que esse tipo de relacionamento não pode ser representado apenas com as chaves das respectivas tabelas. Elas servem como ponte de ligação entre duas entidades. Neste exemplo existem 3: aluno\_curso, professor\_disciplina e disciplina\_area\_conhecimento. Abaixo, observa-se a utilização dessas tabelas por meio dos **inserts e selects** (comandos de seleção) combinados com os **joins** (comandos de combinação).



```

218 -----
219 -- Tabelas associativas (aluno curso
220 -- Aluno Curso
221 -- Rafael Dourado foi matriculado no curso de Des. Web
222 • insert into aluno_curso(id_aluno, id_curso) values (1,1);
223 -- Roberto Santiago foi matriculado no curso de Rel Humanas
224 • insert into aluno_curso(id_aluno, id_curso) values (2,2);
225 -- A seleção abaixo demonstra isso:
226 • select p.nome as nome_aluno, c.nome as curso_matriculado
227 from aluno a
228 inner join pessoa p on p.id_pessoa = a.id_pessoa
229 inner join aluno_curso ac on ac.id_aluno = a.id_aluno
230 inner join curso c on c.id_curso = ac.id_curso;

```



Result Grid



Filter Rows:

Export:





Wrap Cell Content:

	nome_aluno	curso_matriculado
▶	Rafael Dourado	Desenvolvimento-Web
	Roberto Santiago	Relações Humanas


```
inner join curso c on c.id_curso = ac.id_curso;
-- DQL
-- Inner Join -- Selecionar nome do aluno e matrícula.
select p.nome as nome_aluno, a.matricula, a.id_aluno
from aluno a
inner join pessoa p on p.id_pessoa = a.id_pessoa;
```

Result Grid

Filter Rows:

Export:



	nome_aluno	matricula	id_aluno
▶	Rafael Dourado	GC-20245830	1
	Roberto Santiago	GC-20245831	2

```
238      -- Professor Disciplina
239 •   insert into professor_disciplina(id_professor, id_disciplina) values (1,2),
240      (2,1);
241      -- Percebe-se que Marcos Franco leciona TI e Thaisa Leciona L. Portuguesa
242 •   select p.nome as nome_professor, d.nome as nome_disciplina
243
244      from professor prof
245      inner join professor_disciplina pd on pd.id_professor = prof.id_professor
246      inner join disciplina d on d.id_disciplina = pd.id_disciplina
247      inner join pessoa p on prof.id_pessoa = p.id_pessoa;
248
```

<   Filter Rows:  | Export:  | Wrap Cell Content: 

	nome_professor	nome_disciplina
▶	Marcos Franco	Tecnologia da Informação
	Thaisa Leal	Lingua Portuguesa

```

251 -- Disciplinas cadastradas no curso
252 insert into curso_disciplina(id_curso, id_disciplina) values (1,2), (1,5);
253
254 -- Percebe-se duas disciplinas cadastradas no curso de Des. Web:
255
256 • select c.nome as nome_curso, d.nome as nome_disciplina
257
258 from curso c
259 inner join curso_disciplina cd on cd.id_curso = c.id_curso
260 inner join disciplina d on d.id_disciplina = cd.id_disciplina;

```



Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	nome_curso	nome_disciplina
▶	Desenvolvimento-Web	Tecnologia da Informação
	Desenvolvimento-Web	Matemática Básica

# Ponto Principal.

Finaliza-se o bd Ciclos de Estudo com a implementação e uso da entidade `ciclo_estudo`, assim representa-se o uso da regra de negócio para registrar ciclos de estudo.



```
-- Ciclo de Estudo
-- Por fim o topico principal do escopo do projeto: Cadastrar ciclos de estudo
-- no insert não é necessário cadastrar o tempo do ciclo pois ele foi automatizado pela trigger abaixo:
DELIMITER //
CREATE TRIGGER subtracao_tempo
BEFORE INSERT ON ciclo_estudo
FOR EACH ROW
BEGIN
    SET NEW.tempo_ciclo = NEW.fim_ciclo - NEW.inicio_ciclo;
END;
//
```





```

279 -- O exemplo de junção abaixo seleciona o assunto de acordo com a disciplina, o curso e mostra o tempo
280 -- gasto durante o ciclo de estudo.
281
282 • select d.nome as diciplina, ass.nome as assunto_estudado, p.nome as professor,
283 c.nome as curso, ce.tempo_ciclo as tempo_estudo, ce.tempo_liquido
284
285 from ciclo_estudo ce
286 inner join assunto ass on ass.id_assunto = ce.id_assunto
287 inner join curso c on c.id_curso = ce.id_curso
288 inner join disciplina d on d.id_disciplina = ass.id_disciplina
289 inner join professor_disciplina pd on pd.id_disciplina = d.id_disciplina
290 inner join professor prof on prof.id_professor = pd.id_professor
291 inner join pessoa p on p.id_pessoa = prof.id_pessoa;
292

```

<

Result Grid   Filter Rows:  Export:  Wrap Cell Content: 

	diciplina	assunto_estudado	professor	curso	tempo_estudo	tempo_liquido
▶	Raciocínio Lógico	Lógica de primeira ordem!	Marcos Franco	Desenvolvimento-Web	00:28:00	00:28:00
	Raciocínio Lógico	Lógica de primeira ordem!	Marcos Franco	Desenvolvimento-Web	00:28:00	00:28:00
	Direito Constitucional	Direitos e deveres individuais e coletivos	Maicon	Relações Humanas	00:00:00	00:00:00
	Tecnologia da Informação	Pogramação Java	Marcos Franco	Desenvolvimento-Web	00:10:00	00:10:00
	Lingua Portuguesa	comunicação formal	Thaia Leal	Relações Humanas	00:15:00	00:15:00



# Tabela ‘Pausa’.

Esta tabela serve para registrar as interrupções do aluno, subtrair essas interrupções do ciclo e calcular o resultado no campo tempo líquido.

```
297 DELIMITER //
```

298 • CREATE TRIGGER tempo\_pausa

```
299 BEFORE INSERT ON pausa
300 FOR EACH ROW
301 BEGIN
302     SET NEW.tempo_pausa = NEW.fim_pausa - NEW.inicio_pausa;
303 END;
304 //
```

Result Grid



Filter Rows:

Edit:



Export/Import:









	id_pausa	inicio_pausa	fim_pausa	tempo_pausa	id_ciclo_estudo
*	NULL	NULL	NULL	NULL	NULL

```

308 DELIMITER //
309 • CREATE TRIGGER atualiza_tempo_liquido
310 AFTER INSERT ON pausa
311 FOR EACH ROW
312 BEGIN
313     DECLARE soma_tempo_pausa TIME;
314     DECLARE id_ciclo INT;
315     -- Calcula a soma dos tempos de pausa para a chave estrangeira inserida
316     SELECT SUM(tempo_pausa) INTO soma_tempo_pausa
317     FROM pausa
318     WHERE id_ciclo_estudo = NEW.id_ciclo_estudo;
319     -- Subtrai o tempo total de pausa do tempo_ciclo na tabela ciclo_estudo
320     SELECT tempo_ciclo INTO id_ciclo
321     FROM ciclo_estudo
322     WHERE id_ciclo_estudo = NEW.id_ciclo_estudo;
323     -- Atualiza o tempo_ciclo na tabela ciclo_estudo com a subtração do tempo de pausa

```

Result Grid								
Filter Rows: <input type="text"/>								
Edit:   								
Export/Import:  								
Wrap Cell Content: 								
	id_ciclo_estudo	data	inicio_ciclo	fim_ciclo	tempo_ciclo	id_curso	id_assunto	tempo_liquido
▶	1	2023-12-15	09:30:00	09:58:00	00:28:00	1	1	00:28:00
	2	2023-12-15	09:30:00	09:58:00	00:28:00	1	1	00:28:00
	3	2023-12-15	08:30:00	09:10:00	00:00:00	2	2	00:00:00
	4	2023-12-20	13:00:00	13:10:00	00:10:00	1	4	00:10:00
	5	2023-12-20	13:15:00	13:30:00	00:15:00	2	5	00:15:00

```
DELIMITER //
```

```
CREATE TRIGGER atualiza_tempo_liquido
```

```
AFTER INSERT ON pausa
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    DECLARE soma_tempo_pausa TIME;
```

```
    DECLARE id_ciclo INT;
```

```
    -- Calcula a soma dos tempos de pausa para a chave estrangeira inserida
```

```
    SELECT SUM(tempo_pausa) INTO soma_tempo_pausa
```

```
    FROM pausa
```

```
    WHERE id_ciclo_estudo = NEW.id_ciclo_estudo;
```

```
    -- Subtrai o tempo total de pausa do tempo_ciclo na tabela ciclo_estudo
```

```
    SELECT tempo_ciclo INTO id_ciclo
```

```
    FROM ciclo_estudo
```

```
    WHERE id_ciclo_estudo = NEW.id_ciclo_estudo;
```

```
    -- Atualiza o tempo_ciclo na tabela ciclo_estudo com a subtração do tempo de pausa
```

```
    UPDATE ciclo_estudo
```

```
    SET tempo_liquido = tempo_ciclo - COALESCE(soma_tempo_pausa, '00:00:00')
```

```
    WHERE id_ciclo_estudo = id_ciclo;
```

```
END;
```

```
//
```

```
DELIMITER ;
```

```
331 • select *from ciclo_estudo;
332
333 -- Cadastro de duas pausas no ciclo de id 1
334 • select *from pausa;
335 • insert into pausa(inicio_pausa, fim_pausa, id_ciclo_estudo)
336 values ('16:38:00', '16:40:00', 1), ('16:38:00', '16:41:00', 1);
337
```



Result Grid



Filter Rows:

Edit:



Export/Import:



Wrap

	id_pausa	inicio_pausa	fim_pausa	tempo_pausa	id_ciclo_estudo
▶	13	16:38:00	16:40:00	00:02:00	1
	14	16:38:00	16:41:00	00:03:00	1
✱	NULL	NULL	NULL	NULL	NULL

331 • `select *from ciclo_estudo;`

332

Result Grid



Filter Rows:

Edit:



Export/Import:



Wrap Cell Content:



	id_ciclo_estudo	data	inicio_ciclo	fim_ciclo	tempo_ciclo	id_curso	id_assunto	tempo_liquido
	1	2023-12-15	09:30:00	09:58:00	00:28:00	1	1	00:23:00
	2	2023-12-15	09:30:00	09:58:00	00:28:00	1	1	00:28:00
	3	2023-12-15	08:30:00	09:10:00	00:00:00	2	2	00:00:00
	4	2023-12-20	13:00:00	13:10:00	00:10:00	1	4	00:10:00
▶	5	2023-12-20	13:15:00	13:30:00	00:15:00	2	5	00:15:00

ciclo\_estudo 75 x

# Conclusão

Portanto é fundamental aplicar, durante o desenvolvimento, todas as fases do banco de dados, desde a análise de requisitos à implementação. Destaca-se a importância da modelagem e diagramação no desenvolvimento dos bancos de dados.

---