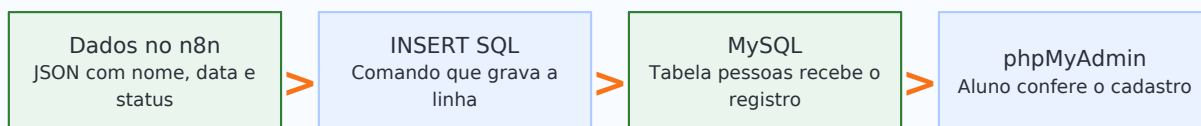


EXECUTE SQL NO N8N

GUIA PRÁTICO DO PRIMEIRO INSERT NA TABELA PESSOAS

Como cadastrar seu primeiro registro no MySQL usando o n8n e o node Execute SQL Query



Objetivo do material

Entender como os dados saem do n8n em formato JSON, entram em uma consulta INSERT INTO e são gravados na tabela pessoas do MySQL.

Material de apoio - SQL PRÓ para n8n

1. O que você vai aprender

Até aqui, o aluno já visualizou a estrutura da tabela pessoas. Agora começa uma etapa essencial: gravar o primeiro registro real no banco de dados.

Consultar dados é importante, mas para alimentar um banco de dados precisamos aprender a inserir novas informações. No MySQL, o comando responsável por isso é o INSERT.

Ideia central

O INSERT é o comando SQL usado para gravar uma nova linha dentro de uma tabela.

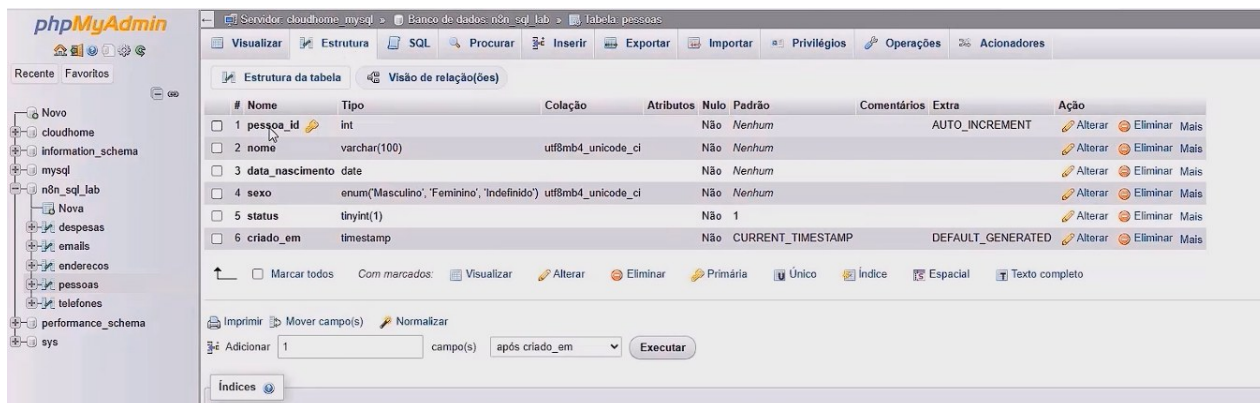
Neste guia, você verá como preparar dados no n8n, montar a consulta SQL, executar o node Execute SQL Query e confirmar o resultado no phpMyAdmin.



Etapa	O que acontece	Ferramenta
1	Montamos os dados da pessoa	n8n
2	Criamos a consulta INSERT	Execute SQL Query
3	O MySQL grava uma nova linha	MySQL
4	Conferimos o cadastro inserido	phpMyAdmin

2. Relembrando a estrutura da tabela pessoas

Antes de montar qualquer INSERT, o primeiro passo é observar a estrutura da tabela. É nela que descobrimos quais campos existem, quais são obrigatórios e quais podem ser preenchidos automaticamente pelo banco.



Campo	Tipo	Informar no INSERT?	Motivo
pessoa_id	INT AUTO_INCREMENT	Não	O MySQL gera o identificador automaticamente.
nome	VARCHAR(100)	Sim	É o nome da pessoa que será cadastrada.
data_nascimento	DATE	Sim	Deve seguir o formato YYYY-MM-DD.
sexo	ENUM	Sim	Precisa respeitar as opções permitidas.
status	TINYINT(1)	Sim ou opcional	No projeto, 1 representa pessoa ativa.
criado_em	TIMESTAMP	Não	Pode ser preenchido com CURRENT_TIMESTAMP.

Regra prática

Nem todo campo da tabela precisa aparecer no INSERT. Campos automáticos, como `pessoa_id` e `criado_em`, podem ser preenchidos pelo próprio banco.

3. O que é o comando INSERT

O INSERT é usado para inserir novos registros em uma tabela. Pense na tabela como uma planilha: cada registro é uma nova linha. O INSERT é o comando que adiciona essa linha.

ESTRUTURA BÁSICA

```
INSERT INTO nome_da_tabela (campo1, campo2, campo3)
VALUES (valor1, valor2, valor3);
```

Parte	Significado
INSERT INTO	Indica que queremos inserir um novo registro.
nome_da_tabela	Mostra em qual tabela o registro será salvo.
(campo1, campo2, campo3)	Lista os campos que receberão valores.
VALUES	Indica os valores que serão gravados.
(valor1, valor2, valor3)	Valores que correspondem aos campos informados.

Atenção

A ordem dos valores precisa seguir a mesma ordem dos campos. Se o primeiro campo é nome, o primeiro valor precisa ser o nome.

No primeiro cadastro da tabela pessoas, vamos informar apenas os campos necessários: nome, data_nascimento, sexo e status.

4. Primeiro INSERT com valores fixos

Antes de usar variáveis do n8n, é útil entender a consulta com valores escritos diretamente no SQL.

SQL COM VALORES FIXOS

```
INSERT INTO pessoas (nome, data_nascimento, sexo, status)
VALUES ('Lucas Andrade Silva', '1992-03-15', 'Masculino', 1);
```

Campo	Valor gravado	Observação
nome	Lucas Andrade Silva	Texto, por isso fica entre aspas simples.
data_nascimento	1992-03-15	Data no padrão do MySQL: YYYY-MM-DD.
sexo	Masculino	Valor permitido pelo ENUM.
status	1	Número indicando pessoa ativa no projeto.

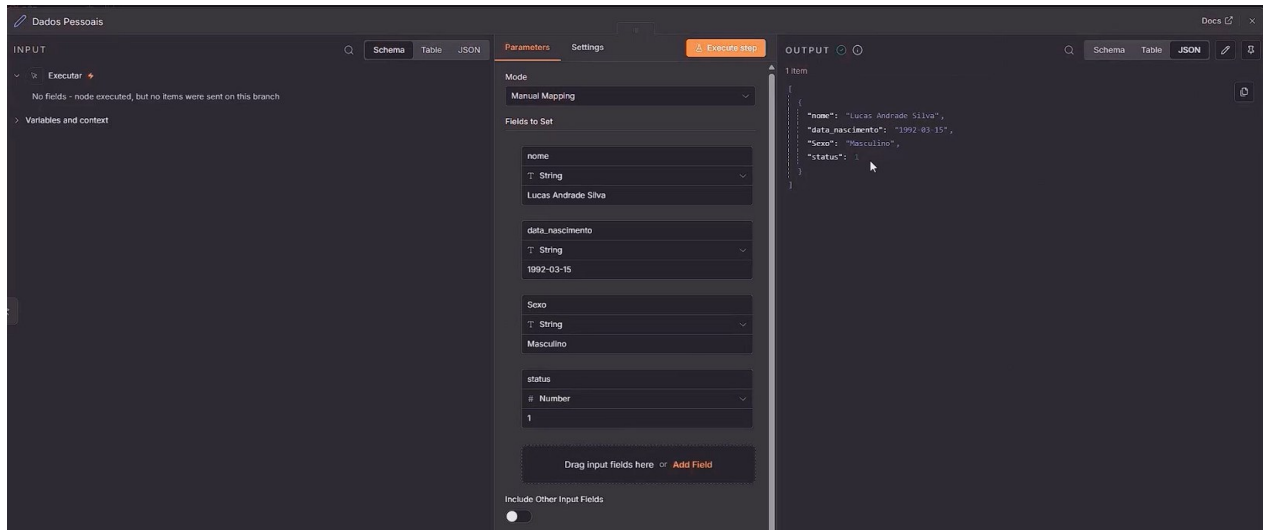
Para entender

Esse modelo fixo é ótimo para aprender a lógica. Porém, em uma automação real, queremos que os valores venham dinamicamente do fluxo do n8n.

Por isso, depois de entender a estrutura básica, substituímos os valores fixos pelas expressões do n8n, como `{{ $json.nome }}`.

5. Preparando os dados no n8n

No n8n, os dados normalmente trafegam em formato JSON. No exemplo da aula, o node Dados Pessoais monta os dados que serão usados no INSERT.



JSON GERADO NO N8N

```
{
  "nome": "Lucas Andrade Silva",
  "data_nascimento": "1992-03-15",
  "Sexo": "Masculino",
  "status": 1
}
```

Campo no JSON	Tipo de informação	Uso no INSERT
nome	Texto	Será gravado no campo nome.
data_nascimento	Data	Será gravada no campo data_nascimento.
Sexo	Texto	Será gravado no campo sexo.
status	Número	Será gravado no campo status.

6. Montando o INSERT com variáveis do n8n

Agora a consulta passa a usar os valores vindos do JSON. Assim, quando o fluxo receber outra pessoa, a mesma query poderá cadastrar outro registro automaticamente.

SQL COM VARIÁVEIS DO N8N

```
INSERT INTO pessoas (nome, data_nascimento, sexo, status)
VALUES (
  '{{ $json.nome }}',
  '{{ $json.data_nascimento }}',
  '{{ $json.Sexo }}',
  {{ $json.status }}
);
```

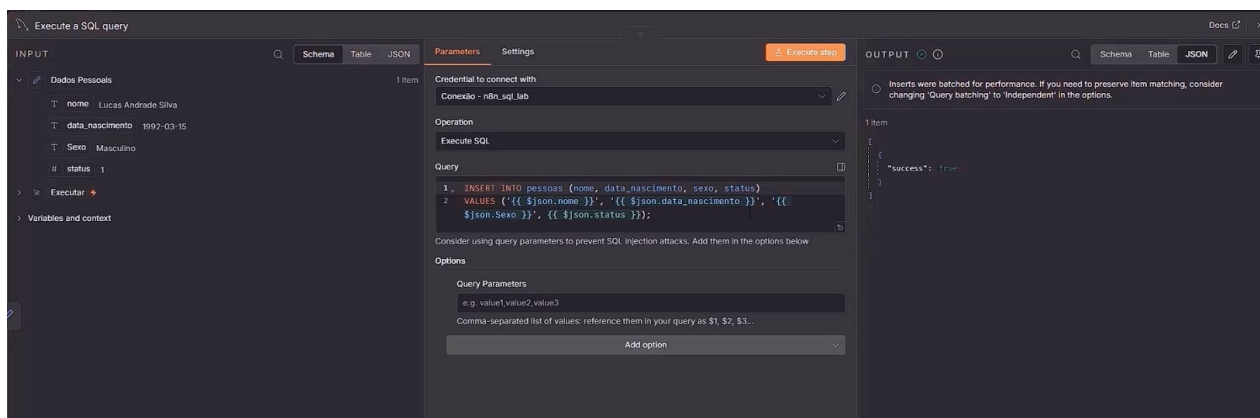
Campo da tabela	Valor usado no n8n	Explicação
nome	{{ \$json.nome }}	Pega o nome que veio no JSON.
data_nascimento	{{ \$json.data_nascimento }}	Pega a data de nascimento.
sexo	{{ \$json.Sexo }}	Pega o valor do campo Sexo.
status	{{ \$json.status }}	Pega o número do status.

Aspas simples

Textos e datas devem ficar entre aspas simples. Por isso usamos '{{ \$json.nome }}', '{{ \$json.data_nascimento }}' e '{{ \$json.Sexo }}'. O status é numérico, então pode ficar sem aspas.

7. Executando o INSERT no node Execute SQL Query

Com os dados preparados e a query montada, o próximo passo é executar o comando no node Execute SQL Query.



Passo	O que fazer
1	Abra o node Execute SQL Query.
2	Selecione a credencial de conexão com o banco n8n_sql_lab.
3	Escolha a operação Execute SQL.
4	Cole a consulta INSERT no campo Query.
5	Verifique se os dados do node anterior chegaram corretamente.
6	Execute o node e confira o OUTPUT.

RESPOSTA DE SUCESSO

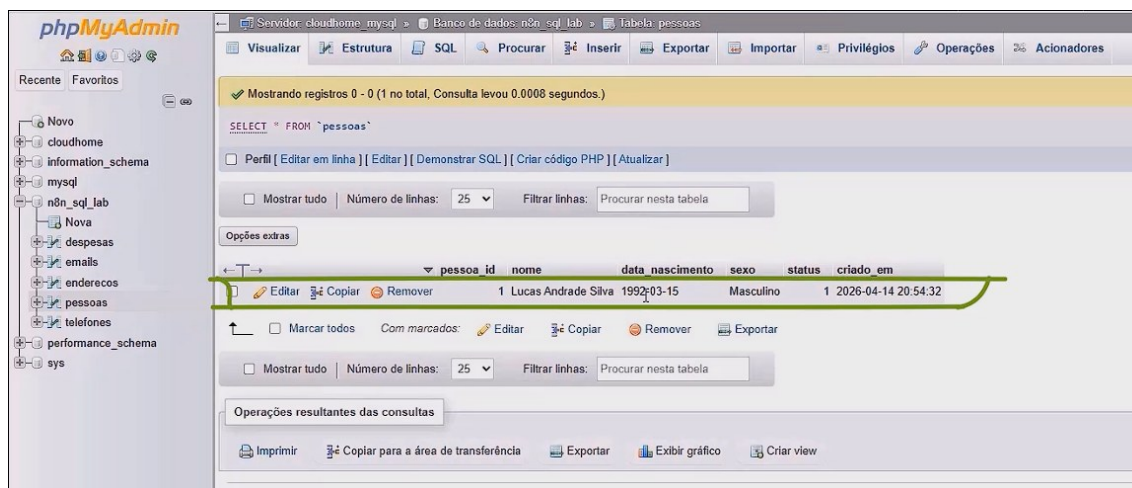
```
{
  "success": true
}
```

Como interpretar

A resposta success: true indica que o comando foi enviado e executado com sucesso no banco.

8. Conferindo o registro no phpMyAdmin

Depois de executar o INSERT no n8n, o teste só fica completo quando conferimos no phpMyAdmin se o registro realmente apareceu na tabela pessoas.



Campo	Valor esperado
pessoa_id	1 - gerado automaticamente pelo MySQL
nome	Lucas Andrade Silva
data_nascimento	1992-03-15
sexo	Masculino
status	1
criado_em	data e hora geradas pelo banco

Conferência final

Se o registro apareceu no phpMyAdmin, significa que o n8n conseguiu enviar os dados e o MySQL gravou a nova linha corretamente.

9. Por que pessoa_id e criado_em ficam fora do INSERT

Dois campos da tabela pessoas não precisam ser informados manualmente no primeiro INSERT: pessoa_id e criado_em.

pessoa_id - AUTO_INCREMENT

É a chave primária da tabela. O MySQL cria o próximo valor vazio, o primeiro registro normalmente recebe pessoa_id = 1.

criado_em - CURRENT_TIMESTAMP

Registra automaticamente a data e hora de criação do cadastro. O banco grava quando o registro foi criado.

Campo	Quem preenche?	Por quê?
pessoa_id	MySQL	Porque está configurado como AUTO_INCREMENT.
criado_em	MySQL	Porque usa CURRENT_TIMESTAMP como padrão.
nome	Aluno/n8n	É uma informação específica da pessoa.
data_nascimento	Aluno/n8n	Precisa vir do cadastro da pessoa.
sexo	Aluno/n8n	Precisa respeitar os valores permitidos no ENUM.
status	Aluno/n8n	Define se a pessoa está ativa no projeto.

Resumo

No INSERT, informe apenas aquilo que precisa ser decidido pelo fluxo. O que o banco consegue gerar automaticamente pode ficar fora da query.

10. Entendendo a query linha por linha

QUERY FINAL

```
INSERT INTO pessoas (nome, data_nascimento, sexo, status)
VALUES (
  '{{ $json.nome }}',
  '{{ $json.data_nascimento }}',
  '{{ $json.Sexo }}',
  {{ $json.status }}
);
```

Linha/trecho	Explicação simples
INSERT INTO pessoas	Insira um novo registro na tabela pessoas.
(nome, data_nascimento, sexo, status)	Esses são os campos que receberão valores.
VALUES	Agora vêm os valores que serão gravados.
'{{ \$json.nome }}'	Pega o nome do JSON e grava no campo nome.
'{{ \$json.data_nascimento }}'	Pega a data do JSON e grava no campo data_nascimento.
'{{ \$json.Sexo }}'	Pega o sexo do JSON e grava no campo sexo.
{{ \$json.status }}	Pega o status numérico do JSON e grava no campo status.

Detalhe importante

No n8n, o nome do campo precisa bater com o nome que aparece no JSON. Se o JSON mostra Sexo com S maiúsculo, use {{ \$json.Sexo }}.

11. Erros comuns no primeiro INSERT

Alguns erros são muito comuns quando o aluno executa seu primeiro INSERT. Veja os principais e como evitar.

Erro 1 - esquecer aspas em textos e datas

Textos e datas precisam ficar entre aspas simples. Números, como status, podem ficar sem aspas.

ERRADO

```
VALUES (Lucas Andrade Silva, 1992-03-15, Masculino, 1);
```

CORRETO

```
VALUES ('Lucas Andrade Silva', '1992-03-15', 'Masculino', 1);
```

Erro 2 - usar data no formato brasileiro

Campos DATE no MySQL devem usar o padrão YYYY-MM-DD.

ERRADO

```
VALUES ('Lucas Andrade Silva', '15/03/1992', 'Masculino', 1);
```

CORRETO

```
VALUES ('Lucas Andrade Silva', '1992-03-15', 'Masculino', 1);
```

12. Mais cuidados antes de executar

Erro 3 - usar valor diferente do ENUM

O campo sexo aceita apenas os valores definidos no ENUM: Masculino, Feminino ou Indefinido.

ERRADO

```
VALUES ('Lucas Andrade Silva', '1992-03-15', 'Homem', 1);
```

CORRETO

```
VALUES ('Lucas Andrade Silva', '1992-03-15', 'Masculino', 1);
```

Erro 4 - trocar a ordem dos campos e valores

A ordem dos valores precisa corresponder à ordem dos campos.

ERRADO

```
INSERT INTO pessoas (nome, data_nascimento, sexo, status)
VALUES ('1992-03-15', 'Lucas Andrade Silva', 'Masculino', 1);
```

CORRETO

```
INSERT INTO pessoas (nome, data_nascimento, sexo, status)
VALUES ('Lucas Andrade Silva', '1992-03-15', 'Masculino', 1);
```

Boas práticas

Confira a estrutura da tabela, identifique os campos automáticos, respeite o formato das datas, valide o JSON de entrada e sempre confirme no phpMyAdmin se o registro foi inserido.

13. Checklist antes de executar um INSERT

☐ Estou usando a tabela correta?	☐ Textos e datas estão entre aspas simples?
☐ Conferi a estrutura da tabela pessoas?	☐ O status está correto?
☐ Sei quais campos são automáticos?	☐ Os nomes do JSON estão iguais aos campos do n8n?
☐ Deixei pessoa_id fora do INSERT?	☐ A ordem dos campos corresponde à ordem dos valores?
☐ Deixei criado_em fora do INSERT?	☐ Executei a query no node Execute SQL Query?
☐ Informe nome, data_nascimento, sexo e status?	☐ Conferi se apareceu success true?
☐ A data está no formato YYYY-MM-DD?	☐ Verifiquei no phpMyAdmin se o registro foi inserido?
☐ O valor de sexo respeita o ENUM?	

Sequência recomendada

Primeiro entenda a tabela. Depois monte o INSERT. Depois execute no n8n. Por fim, confira no phpMyAdmin.

Este checklist pode ser usado em qualquer INSERT futuro, não apenas na tabela pessoas. A lógica é sempre a mesma: campos corretos, valores corretos, ordem correta e conferência final.

14. Resumo rápido - cola de consulta

Comando ou conceito	Para que serve	Exemplo
INSERT INTO	Informa a tabela e os campos	INSERT INTO pessoas (...)
VALUES	Informa os valores	VALUES (...)
AUTO_INCREMENT	Gera o ID automaticamente	pessoa_id
CURRENT_TIMESTAMP	Gera data e hora automaticamente	criado_em
DATE	Campo de data	1992-03-15
ENUM	Campo com opções limitadas	Masculino, Feminino, Indefinido
{{ \$json.nome }}	Pega valor do JSON no n8n	Lucas Andrade Silva

MODELO COM VALORES FIXOS

```
INSERT INTO pessoas (nome, data_nascimento, sexo, status)
VALUES ('Lucas Andrade Silva', '1992-03-15', 'Masculino', 1);
```

MODELO COM VARIÁVEIS DO N8N

```
INSERT INTO pessoas (nome, data_nascimento, sexo, status)
VALUES (
  '{{ $json.nome }}',
  '{{ $json.data_nascimento }}',
  '{{ $json.Sexo }}',
  {{ $json.status }}
);
```

Como usar

Use o modelo com valores fixos para entender. Use o modelo com variáveis do n8n para automatizar.

Encerramento

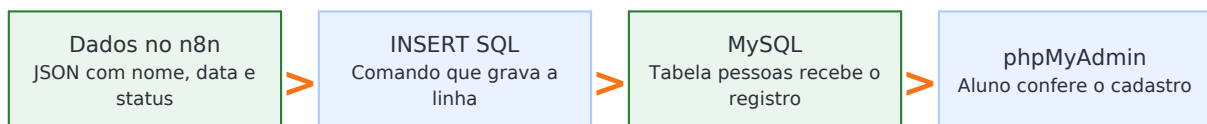
Com o primeiro INSERT, o aluno dá um passo essencial: deixa de apenas consultar dados e começa a alimentar o banco de dados com informações reais.

A partir desse ponto, o n8n passa a ser capaz de enviar dados para o MySQL e construir automações muito mais profissionais.

Esse é o início da integração prática entre workflows, dados e banco de dados relacional.

Mensagem principal

O INSERT transforma dados recebidos no n8n em registros reais dentro do MySQL.



Material de apoio - EXECUTE SQL NO N8N