

## Faça como eu fiz

Chegou a hora de você pôr em prática o que foi visto na aula. Para isso, execute os passos listados abaixo.

---

1) Vamos aprender como manipulamos os Arrays dentro do JSON, no MySQL. Para isso crie uma tabela temporária com o script abaixo:

```
CREATE TABLE t1 (`JSON` json);
```

2) Inclua um JSON com um array nesta tabela:

```
INSERT INTO t1 VALUES ('{"Hobby":["Praia", "Futebol", "Volei"]}');
```

3) Para exibir o conteúdo do campo com o array execute:

```
SELECT JSON_EXTRACT(`JSON`, "$.Hobby") as HOBBIES FROM t1;
```

4) Se quisermos obter apenas o primeiro elemento do array execute:

```
SELECT JSON_EXTRACT(`JSON`, "$.Hobby[0]") as HOBBIES FROM t1;
```

5) Vamos melhorar nossa análise. Iremos apagar a tabela criada anteriormente e recriá-la incluindo um JSON mais complexo;

```
DROP TABLE t1;
CREATE TABLE t1 (`JSON` json);
INSERT INTO t1 VALUES ('{
  "Cliente": [
    {
      "Nome": "Joao",
      "Hobby": [
        "Praia",
        "Futebol",
        "Volei"
      ]
    }
  ]
}');
```

6) Para obter o elemento Nome, do JSON que é um elemento do Array execute:

```
SELECT JSON_EXTRACT(`JSON`, "$.Cliente[0].Nome") FROM t1;
```

7) Vamos deixar nosso JSON mais complexo. Execute:

```
DROP TABLE t1;
CREATE TABLE t1 (`JSON` json);
INSERT INTO t1 VALUES ('{
  "Cliente": [
    {
      "Nome": "Joao",
      "Hobby": [
        "Praia",
        "Futebol",
        "Volei"
      ]
    },
    {
      "Nome": "Carlos",
      "Hobby": [
        "Cinema",
        "Luta"
      ]
    }
  ]
}');
```

8) Para obter o segundo Hobby, do array de Hobbies do segundo cliente do array de Clientes basta executar:

```
SELECT JSON_EXTRACT(`JSON`, "$.Cliente[0].Hobby[1]") FROM t1;
```

9) Nova situação com um JSON com outro formato. Execute:

```
DROP TABLE t1;
CREATE TABLE t1 (`JSON` json);
INSERT INTO t1 VALUES ('{
  "Cliente": [
    {
      "Nome": "Joao",
      "Hobby": "Praia"
    },
    {
      "Nome": "Carlos",
      "Hobby": "Cinema"
    }
  ]
}');
```

10) Se quisermos obter a lista de clientes e hobbies de forma "desnormalizada" basta executar:

```
SELECT JSON_EXTRACT(`JSON`, "$.Cliente[0].Nome") AS NOME, JSON_EXTRACT(`JSON`, "$.Cliente[0].H")
UNION
SELECT JSON_EXTRACT(`JSON`, "$.Cliente[1].Nome") AS NOME, JSON_EXTRACT(`JSON`, "$.Cliente[1].H")
```

11) Mas esta "desnormalização" nos obrigou a usar um Union para cada elemento do Array de Hobbies. Nem sempre iremos saber, de antemão, o número de elementos do Array. Logo podemos refazer a consulta usando JSON\_TABLE. Execute:

```
SELECT t2.NOME, t2.HOBBY FROM t1
CROSS JOIN
JSON_TABLE (JSON_EXTRACT(`JSON`, "$.Cliente"), "$[*]"
COLUMNS (NOME VARCHAR(10) PATH "$.Nome", HOBBY VARCHAR(10) PATH "$.Hobby")) t2;
```

12) Vamos aplicar o JSON\_TABLE para obter a lista dos dependentes associados aos funcionários, na base de dados empresa. Execute:

```
SELECT JSON_EXTRACT (tb_object_funcionario.`JSON`, "$.Cpf") AS Cpf_Funcionario,
tb_object_dependente.Nome_Dependente,
tb_object_dependente.Sexo,
tb_object_dependente.Data_Nascimento,
tb_object_dependente.Parentesco
FROM tb_object_funcionario
CROSS JOIN
JSON_TABLE(JSON_EXTRACT (`JSON`, "$.Dependentes"), "$[*]"
COLUMNS (Nome_Dependente VARCHAR(20) PATH "$.Nome_Dependente",
Sexo VARCHAR(20) PATH "$.Sexo",
Data_Nascimento Datetime PATH "$.Data_Nascimento",
Parentesco VARCHAR(10) PATH "$.Parentesco")) AS tb_object_dependente;
```

13) Podemos repetir o desafio de obter o nome dos funcionários que não possuam dependentes. A solução é a mostrada abaixo:

```
SELECT JSON_EXTRACT(tb_object_funcionario.`JSON`, "$.Primeiro_Nome") AS PRIMEIRO_NOME,
JSON_EXTRACT(tb_object_funcionario.`JSON`, "$.Nome_Meio") AS NOME_MEIO,
JSON_EXTRACT(tb_object_funcionario.`JSON`, "$.Ultimo_Nome") AS ULTIMO_NOME
FROM tb_object_funcionario
LEFT JOIN
(SELECT JSON_EXTRACT (tb_object_funcionario.`JSON`, "$.Cpf") AS Cpf_Funcionario,
tb_object_dependente.Nome_Dependente,
tb_object_dependente.Sexo,
tb_object_dependente.Data_Nascimento,
tb_object_dependente.Parentesco
FROM tb_object_funcionario
CROSS JOIN
JSON_TABLE(JSON_EXTRACT (`JSON`, "$.Dependentes"), "$[*]"
COLUMNS (Nome_Dependente VARCHAR(20) PATH "$.Nome_Dependente",
Sexo VARCHAR(20) PATH "$.Sexo",
Data_Nascimento Datetime PATH "$.Data_Nascimento",
Parentesco VARCHAR(10) PATH "$.Parentesco")) AS tb_object_dependente
) tb_dependente
ON JSON_EXTRACT(tb_object_funcionario.`JSON`, "$.Cpf") = tb_dependente.Cpf_Funcionario
WHERE tb_dependente.Cpf_Funcionario IS NULL;
```

