

ORACLE

MySQL & JavaScript: Perfect Together

A Guide to Using JavaScript to Create Stored
Functions

Scott Stroz

MySQL Developer Advocate



MySQL™

Obligatory "I Love Me" Slide

Full-stack developer before we were called "full-stack" developers

- The only constant in my development stack has been **MySQL**

I used to be a paramedic

- Delivered 3 babies

I have the best office mate!



Obligatory "I Love Me" Slide

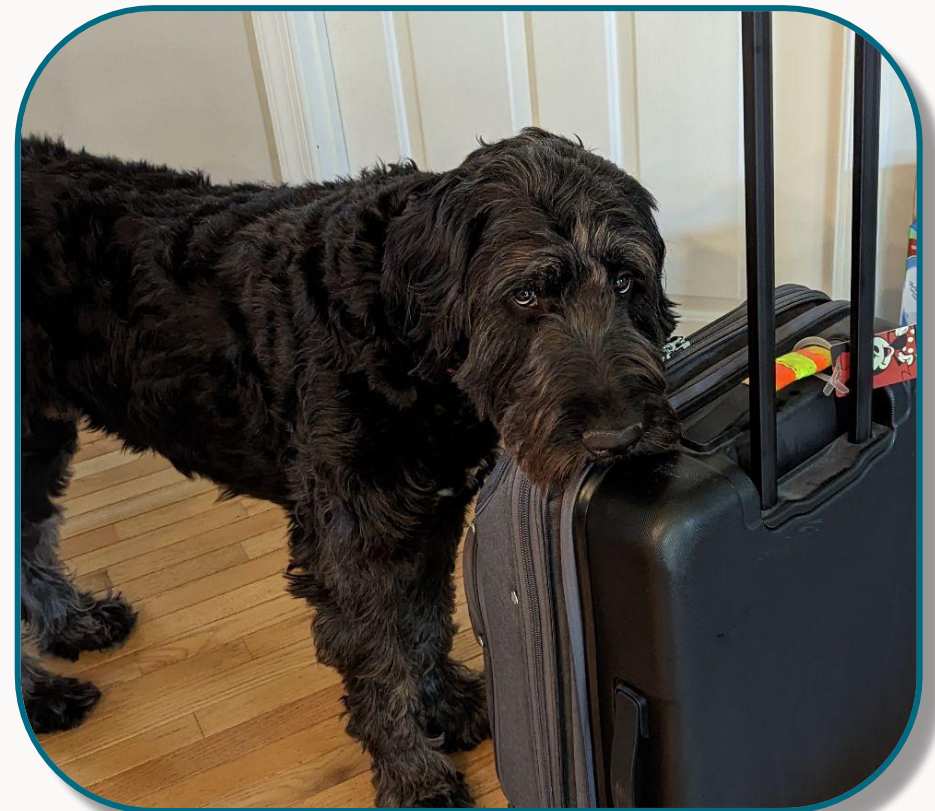
Full-stack developer before we were called "full-stack" developers

- The only constant in my development stack has been **MySQL**

I used to be a paramedic

- Delivered 3 babies

I have the best office mate!



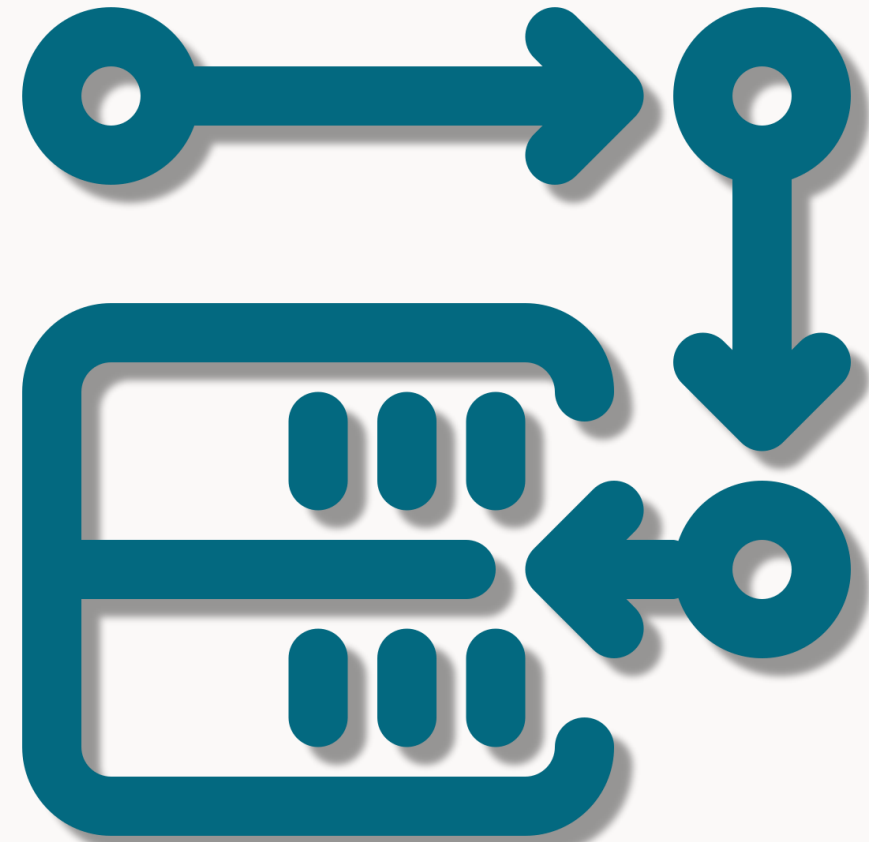
Agenda

- Stored procedures vs. stored functions
- Why JavaScript?
 - Advantages of using JavaScript over SQL
- How does it work?
- Examples



MySQL Stored Procedures

- Prepared SQL code that can be reused
- Often used to encapsulate complex workflows
- Can contain transaction statements
- Cannot be used as part of a query
- Executed using the **CALL** keyword
- Can have defined input and output parameters
- Can return results of a query



MySQL Stored Functions



A large, stylized orange graphic of the mathematical notation $f(x)$. The letter 'f' is tall and slanted, while the 'x' is a simple, bold, blocky character. The parentheses are also thick and rounded.

- Prepared SQL code that can be reused
- Often used to encapsulate complex workflows
- Can be used in a query
 - **SELECT, WHERE, ORDER BY**, etc.
- Can accept arguments
- Can only return simple values
- Are **DETERMINISTIC** or **NOT DETERMINISTIC**

Deterministic vs. Not Deterministic

Deterministic

Always produces the same result given the same parameters

```
DATE_FORMAT("2024-10-10", "%Y")
```

Will **ALWAYS** return 2024

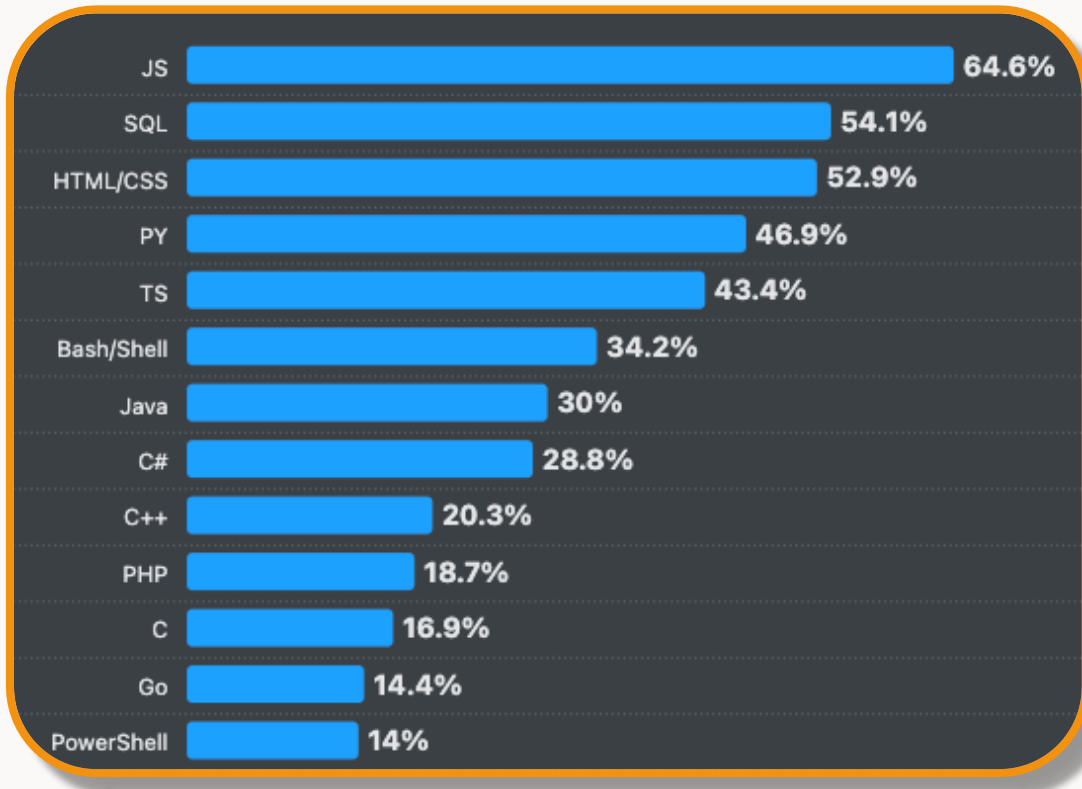
Not Deterministic

Does not always produce the same result given the same parameters

```
NOW()
```

Will rarely (if ever) return the same value

Why JavaScript?



JavaScript is the first external language supported by **MySQL**.

- JavaScript has become ubiquitous
- Large developer base and development ecosystem



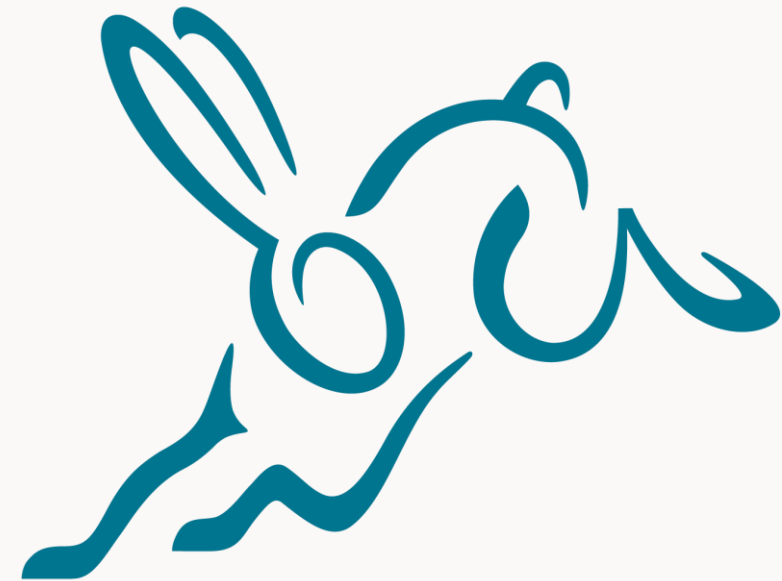
Advantages of Using JavaScript

- In-depth knowledge of SQL is not required
- SQL functionality in stored programs can be limited or overly verbose
- Faster execution with JavaScript
- Better processing of JSON data



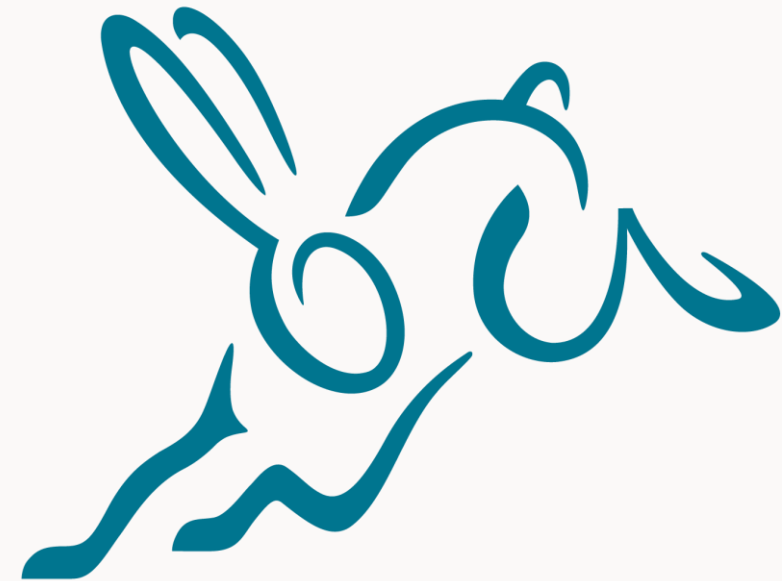
Say Hello to GraalVM!

- GraalVM is a high-performance, polyglot virtual machine
- Supports multiple languages
 - JS, Python, Ruby, and more
- Enhances performance and enables interoperability between languages



Say Hello to GraalVM!

- GraalVM allows us to execute multiple programming languages in a single runtime.
- For MySQL, we can expand the scope of how we execute logic beyond SQL



Examples

Hello Wor..err...New York!!

```
1 CREATE FUNCTION hello_world_js()  
2 RETURNS CHAR(20) DETERMINISTIC 3  
4 LANGUAGE JAVASCRIPT AS $$ 5  
6 return ("Hello New York!!!");  
7  
$;  
;
```

Hello Wor..err...New York!!

```
select hello_world_js();
```

Hello Wor..err...New York!!

```
+-----+  
| hello_world_js() |  
+-----+  
| Hello New York!!! |  
+-----+  
1 row in set (0.0166 sec)
```

Sorting JSON

```
create function sortByProperty(  
  items varchar(10000),  
  prop varchar(100) 1  
  returns varchar(10000) language javascript as $$  
  const arr = JSON.parse(items)  
  function compare(a,b){  
    2 if(a[prop] < b[prop]) return -1  
    else if (a[prop] > b[prop]) return 1  
    return 0  
  }  
  return JSON.stringify(arr.sort(compare)) 3  
  $$;
```

Sorting JSON

```
set @team = '[
  {"name": "Fred", "country": "BE"},
  {"name": "Scott", "country": "US"},
  {"name": "Lenka", "country": "CZ"},
  {"name": "Heather", "country": "US"}
]';

select sortByProperty(@team, 'name') sorted_array;\G
```

Sorting JSON

```
***** 1. row *****  
sorted_array: [  
  {"name": "Fred", "country": "BE"},  
  {"name": "Heather", "country": "US"},  
  {"name": "Lenka", "country": "CZ"},  
  {"name": "Scott", "country": "US"}  
]  
1 row in set (0.0711 sec)
```

Using JavaScript Global Objects - Intl

Episode-001 - Using MySQL Trim() Function
MySQL • 13K views • 2 years ago

Episode-002 - Connecting to a MySQL Database Using MySQL Shell
MySQL • 15K views • 2 years ago

Episode-003 - Connecting to a MySQL Database Over SSL Using MySQL Shell
MySQL • 4.5K views • 2 years ago



Using JavaScript Global Objects - Intl

id	name	score
4	Kate	450912
3	Ryan	345123
2	Jessica	134234
6	Lindy	123987
5	Ray	99876
1	Scott	53976



Using JavaScript Global Objects - Intl

```
create function compactNumberFormat(num int)
  returns varchar(256) language javascript as $$
  return new Intl.NumberFormat('en-US',
    {
      notation: 'compact',
      maximumSignificantDigits: 3
    }).format(num)
  $$;
```

Using JavaScript Global Objects - Intl

```
select
  name,
  score,
  compactNumberFormat(score) compact_score
from scores
order by score desc;
```

Using JavaScript Global Objects - Intl

name	score	compact_score
Kate	45091912	45.1M
Ryan	34512233	34.5M
Jessica	534234	534K
Lindy	423987	424K
Ray	99876	99.9K
Scott	53976	54K

Restrictions – What can't we do?

Interact with
the file system

File System API

Access
network
resources

`fetch()`

Debugging JavaScript in MySQL

```
create function division(a int, b int)
  returns double language javascript as $$
  function validDenominator(num){
    1 console.log("Validating input value: ", num)
    return num !== 0
  }
  return validDenominator(b) ? a/b : null 2
  $$;
```

Debugging JavaScript in MySQL

```
select division(5,2) demo_1, division(2,0) demo_2;
```

Debugging JavaScript in MySQL

demo_1	demo_2
2.5	NULL



Debugging JavaScript in MySQL

```
select mle_session_state("stdout") console\G4
```

Debugging JavaScript in MySQL

```
***** 1. row *****  
console: Validating input value: 2  
Validating input value: 0
```

How to reach me...

- scott.stroz@oracle.com
- BlueSky: [@sstroz](#)
- LinkedIn: [scott-stroz](#)





Thank you!
