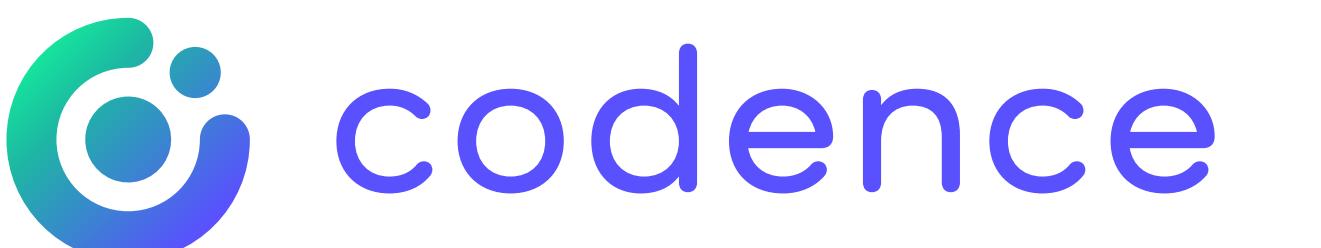


Bearskin Hats

Using Guard Clauses for More Robust Programming

February 8, 3:00 pm–4:00pm

Cristos Lianides-Chin

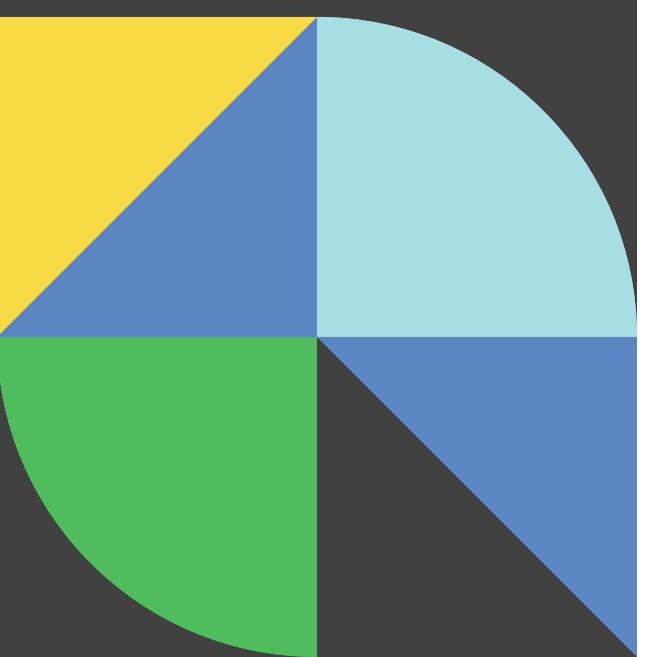


Who Am I?

Platform Architect at Codence

Worked in FileMaker since 2012 (no `fp5` conversions)

Passionate about cooking, sailing, and ~~lazy~~ efficient coding practices



cristos.lianides-chin@codence.com

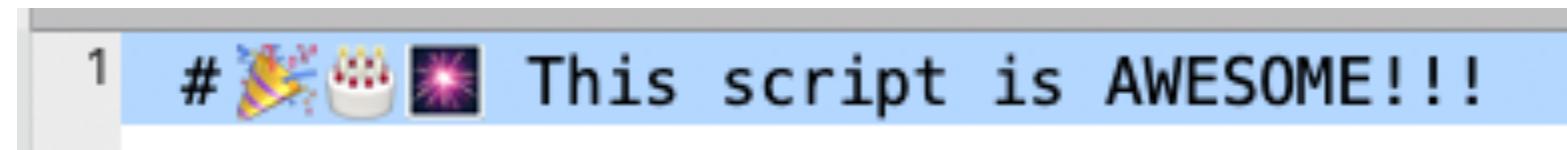
 codence

Take Aways

01

Junior Developers

- You can use emojis in FileMaker script comments



02

Intermediate Developers

- Build a foundational understanding of how to use **guard clauses**
- Learn how to implement the **Try-Catch** pattern in FileMaker.

03

Tech Leads

- Learn to apply guard clauses more effectively, especially in complex scripting scenarios.
- Gain broader knowledge on coding best practices.



What We'll Cover

✓ In Scope

Guard Clauses

Single-Pass Loops / Try-Catch

✗ NOT In Scope

Error Handling

- Robert Naud | Error Handling Revisited (*DIGFM 2023-10*, <https://youtu.be/muhK5AGVb7o>)

Transactions

- Alondra Torres-Navarro & Barbara Cooney | Advanced modern Claris FileMaker transactions (*ENGAGE 2024*)





Guard



```
# Do some stuff  
$var = Get ( ScriptParameter )
```

...

...

```
# Do some more stuff  
Perform Script [ "Profit!" ; $var ]  
Exit Script []
```

(Demo file)



Guard



```
# Do some stuff  
$var = Get ( ScriptParameter )
```

...

```
❗ If [requirementsNotMet]:
```



```
Exit Script []
```

...

```
# Do some more stuff
```

```
Perform Script [ "Profit!" , $var ]
```

(Demo file)

[DEMO]

If vs Guard

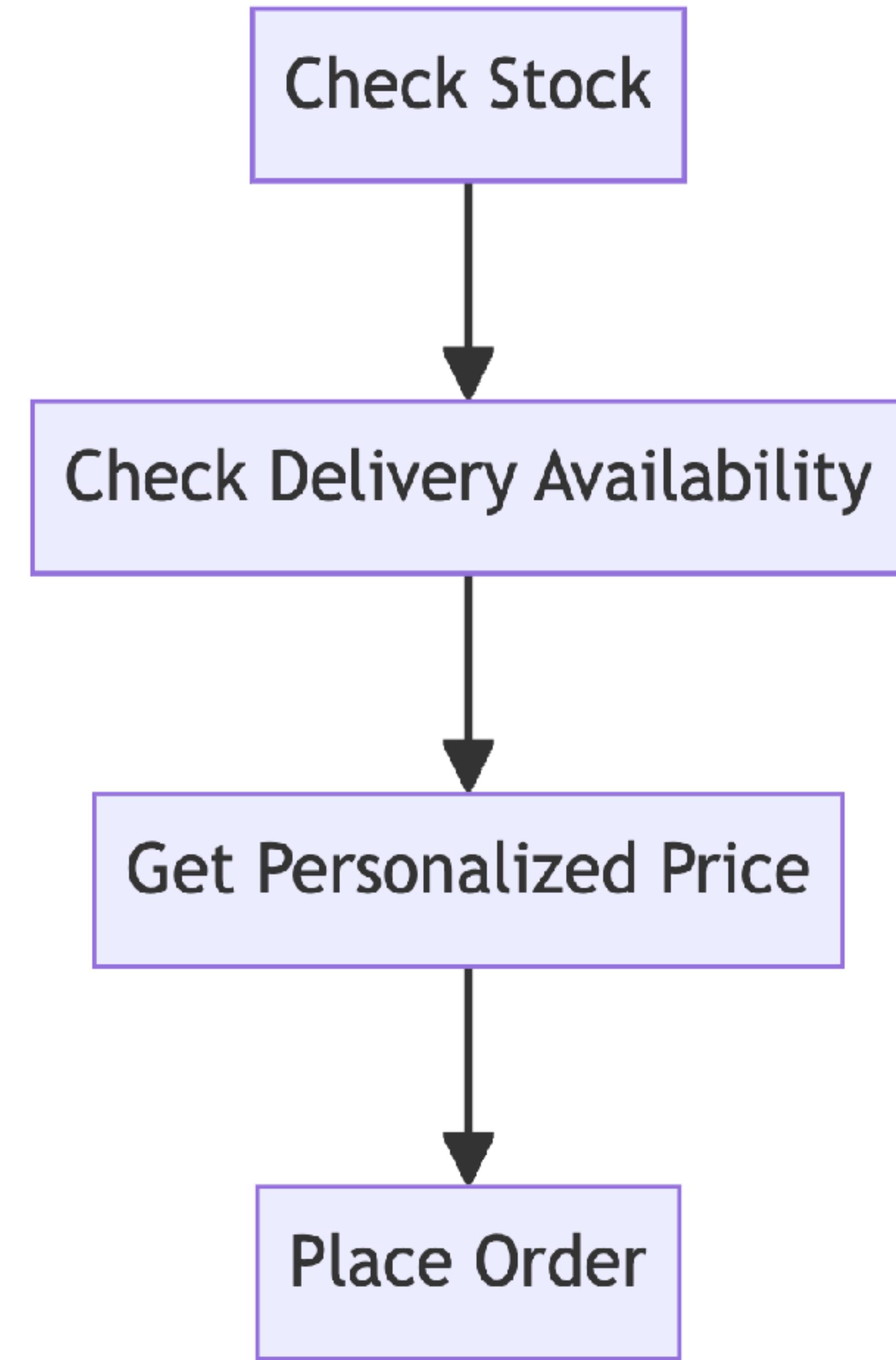
**1 Customer
+ 1 Product
= 1 Order**



(Demo file)

[DEMO]

Order Process



(Demo file)

Why Guard Clauses?

01

More Readable Code

Avoid the Arrow Pattern

02

Fail Fast

Skip to the end when
something goes wrong

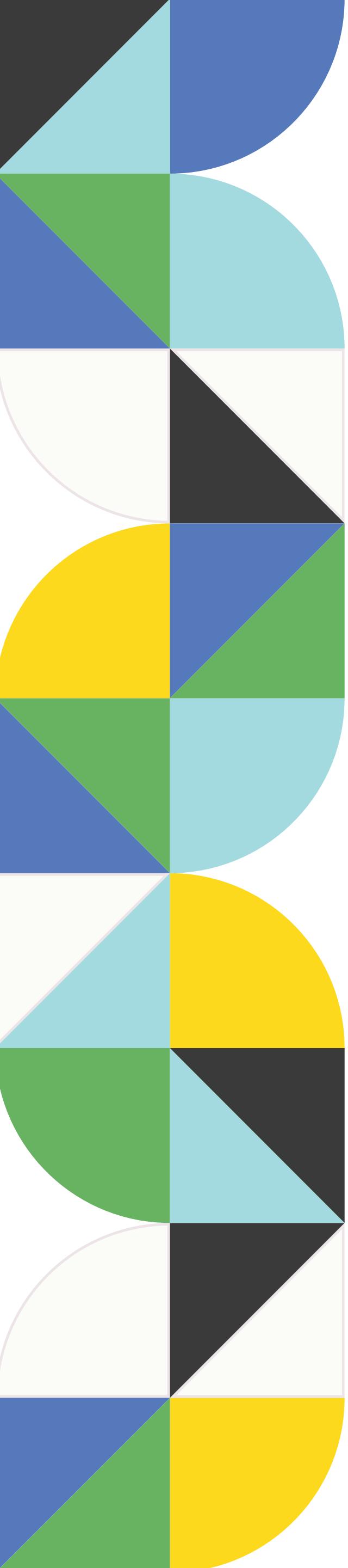
03

Further Reading

- <https://www.codementor.io/@clintwinter/use-guard-clauses-for-cleaner-code-1rrsczgwxp>
- [https://en.wikipedia.org/wiki/Guard_\(computer_science\)](https://en.wikipedia.org/wiki/Guard_(computer_science))

(Demo file)





"It is not enough for code to work. It must also be clean, readable, and understandable."

~ Robert C. Martin, *Clean Code*

[DEMO]
Exit Script
v Exit Loop If



(Demo file)

Try-Catch

Especially Useful in Iterators

```
data = ["10", "20", "a", "30", "40b", "50"]

for item in data:
    try:
        number = int(item) # Attempt to convert string to integer
        print(f"Processed number: {number}")

    except ValueError:
        print(f"Invalid item encountered: '{item}' - Skipping")
```

```
Processed number: 10
Processed number: 20
Invalid item encountered: 'a' - Skipping
Processed number: 30
Invalid item encountered: '40b' - Skipping
Processed number: 50
```

(Demo file)

Single-Pass Loops

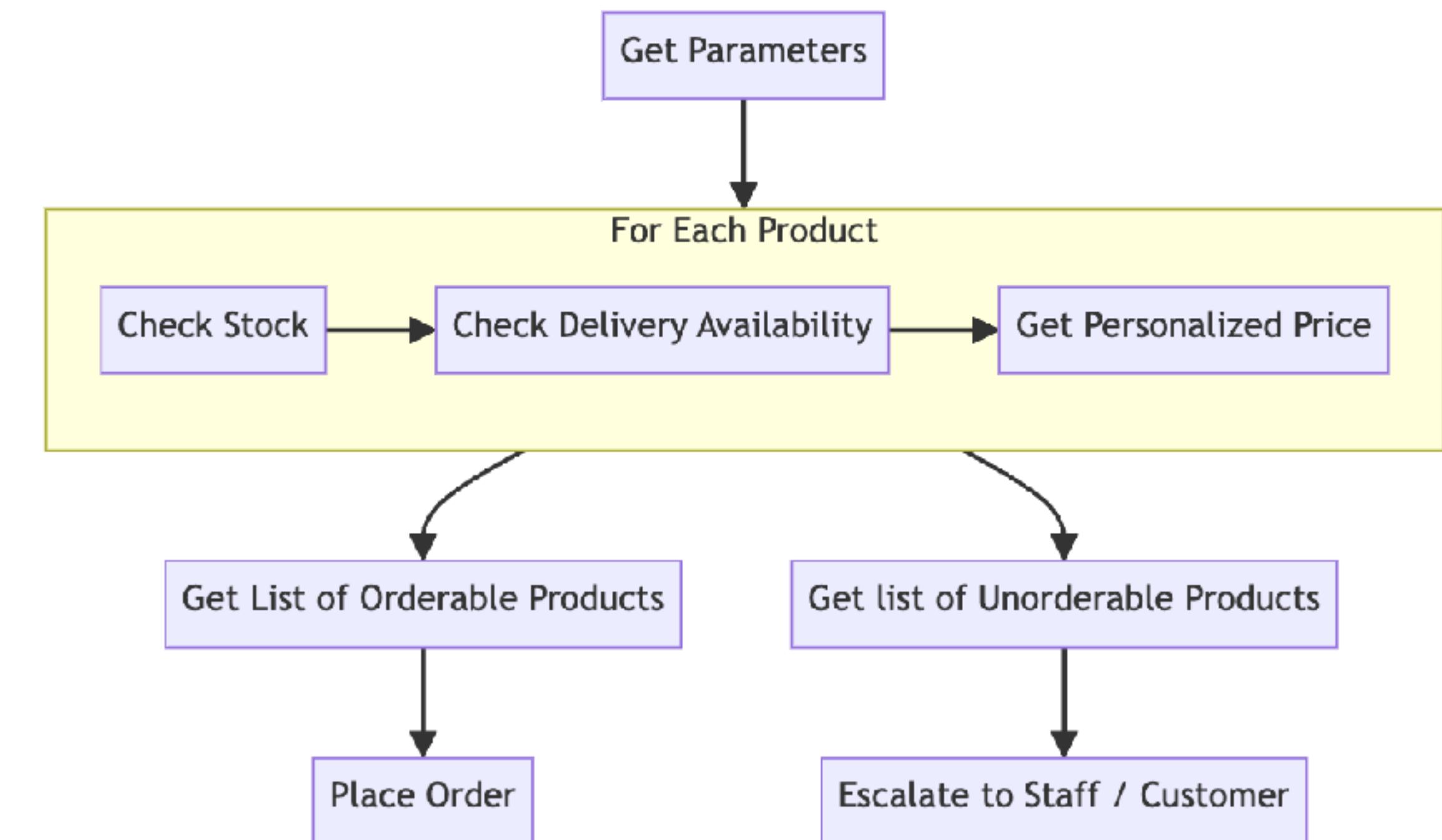
Try-Catch in FileMaker

```
25 # TRY
26 Loop
27 # GUARD
28 Exit Loop If [ x.ifThenThrowException ( value.isNullOrError ( $listSystemTables ) ; "This script requires a l... ]
29
30 # FOREACH ( $tableName in $listSystemTables )
31 Loop
32 Exit Loop If [ Let ( [ $i1Max = ValueCount ( $listSystemTables ) ;$i1 = $i1 + 1 ;$i10 = $i1 - 1 ] ; If ( $... ]
33 Set Variable [ $tableName ; Value: GetValue ( $listSystemTables ; $i1 ) ]
34
35 # INNER TRY
36 Loop
37
38 Go to Layout [ $tableName ; Animation: None ]
39
40 # GUARD
41 Exit Loop If [ x.ifThenThrowException ( not ( Get ( LayoutTableName ) = $tableName ) ; "Expected to be o... ]
42 Exit Loop If [ Get ( TotalRecordCount ) > 0 // SAFE EXIT: Records exist on this table, no need to contin... ]
43
44 New Record/Request
45
46 Exit Loop If [ True //END INNER TRY ]
47 End Loop
48 # END INNER TRY
49
50 End Loop
51 # END FOREACH ( $tableName in $listSystemTables )
52
53 Exit Loop If [ True //END TRY ]
54 End Loop
55
56 # CATCH
57 If [ x.countExceptions ]
58 Set Variable [ $~void ; Value: script.response.addExceptions ]
59 End If
60
61 # FINALLY
62
63 Exit Script [ Text Result: script.response ]
64
```

(Demo file)

[DEMO]

Try-Catch in Other Structures

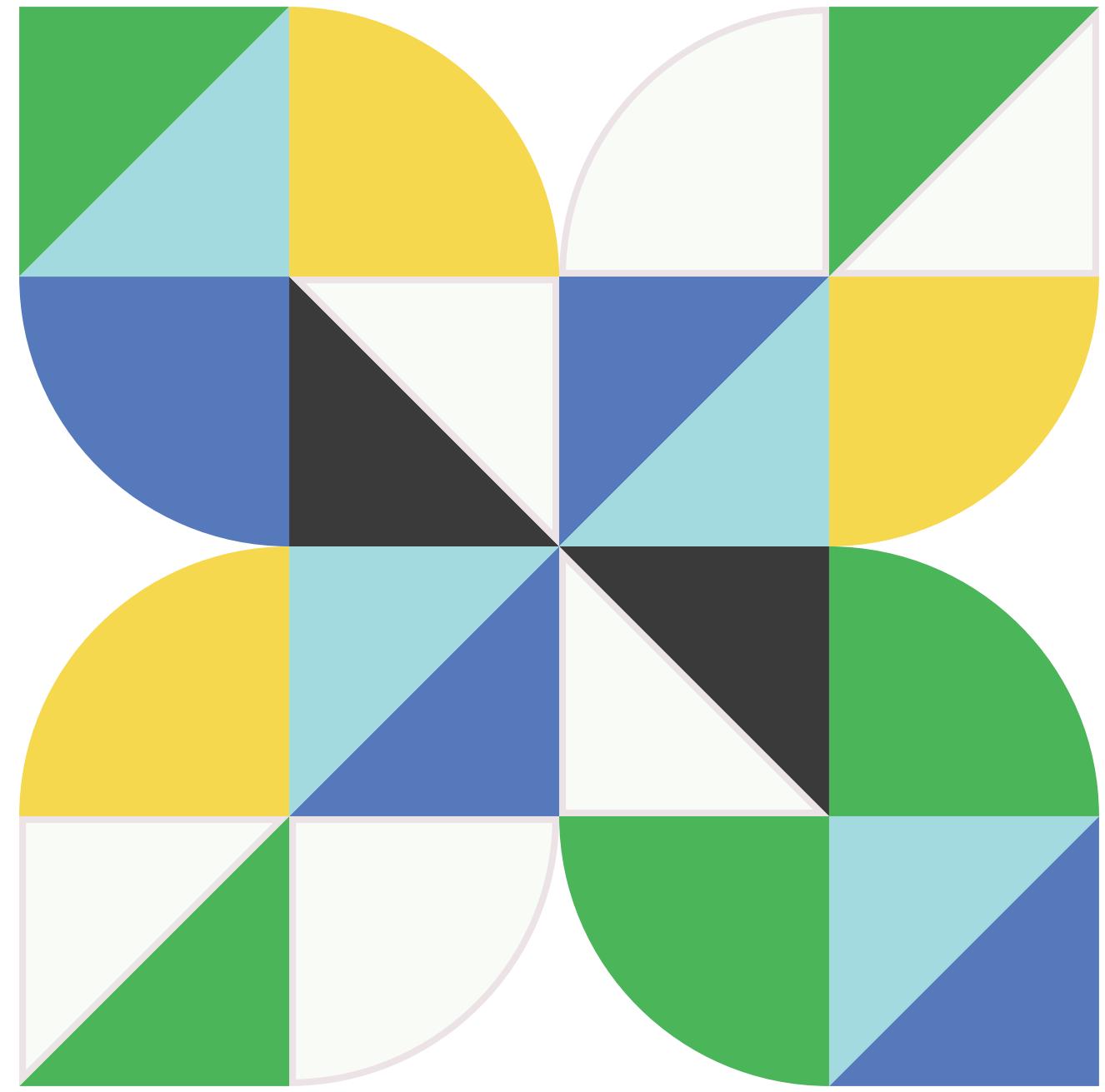


(Demo file)

Scripts vs Calcs

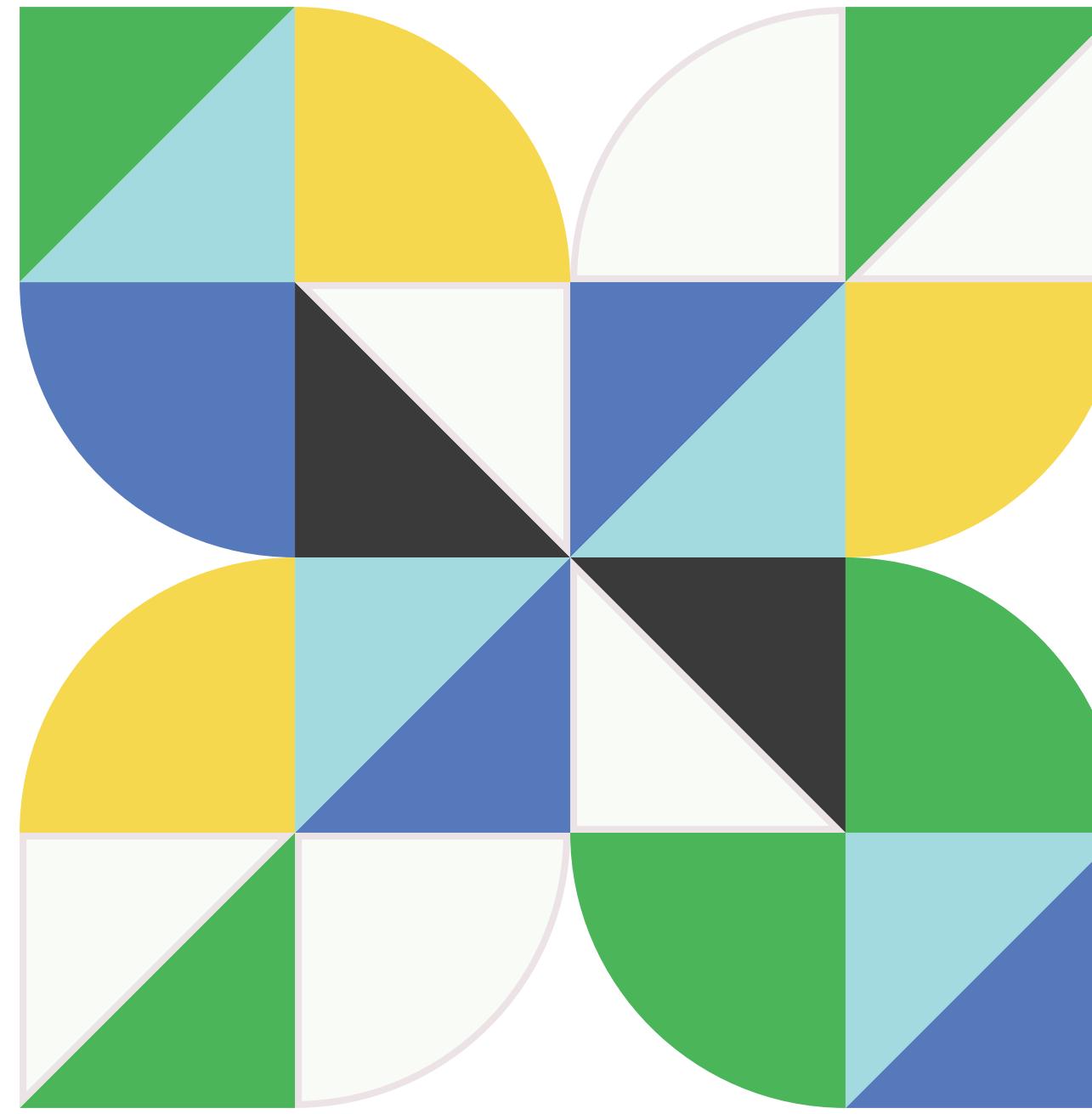
Can't Try-Catch in Calc
Engine

```
22 Let (
23   [ ~parameter = parameter
24
25   ; // Return the result
26   ; ~returnValue = Case
27   ;   ( // Fastest checks go towards the top
28
29   ;     IsEmpty ( parameter ) ;
30   ;       | // Return an error
31   ;         "Error: parameter is empty"
32
33   ;     not json.isValid ( parameter )
34   ;       | "Error: parameter is not valid JSON"
35
36   ;     list.hasValue ( md5.getHash ( ~parameter ) ; $listOtherMd5Hashes )
37   ;       | "Error: parameter occurs multiple times in the set"
38
39   ; // Slower checks go at the bottom
40   ;     not Exact ( ~parameter ; TABLE::Field )
41   ;       | "Error: parameter does not match database value"
42
43   ;     Exact ( ~parameter ; TABLE::Field )
44   ;       | and Exact ( ~parameter ; RelatedTable::Field )
45
46   ; // ELSE (main logic)
47   ;   Let
48   ;     ( // Load data for calc here
49   ;       [ ~fieldData = TABLE::Field
50   ;         ; ~relatedFieldData = RelatedTable::Field
51   ;       ]
52   ;       ; ~parameter & ~fieldData & ~relatedFieldData
53   ;     )
54
55   ;   )
56
57   ;   ~returnValue
58 )
```



Q&A

(Demo file)



How to Type Emoji Comments

(Demo file)

Thank you!

cristos.lianides-chin@codence.com

<https://linkedin.com/in/cristoslc>



Claris Engage 2024