FormatException

Workshop #02

Flutter Developer Bootcamp

# **Purpose**

In this workshop demonstrate how to use try and catch blocks in Dart to handle exceptions, specifically FormatException in this case, that might occur during the parsing of a string to an integer. It helps learners understand the concept of error handling and how to deal with errors that might arise during program execution.

# **Problem**

In this workshop there will be a string 'Empetz' into an integer using int.parse(). It employs a try-catch block to handle the potential FormatException that arises from this operation. If parsing fails, the caught exception is printed, preventing the program from crashing. This error-handling approach ensures the program's stability when dealing with unexpected input.

# **How to Solve**

1. Checkout the workshop from Git Repo:

git clone -b <user-branch> <repo-URL>

2. Open the root folder inside VS Code

3. Open the root folder in terminal

4. Run the command dart run filename.dart

5. Set a FormatException and handle the error

# **You Will Achieve**

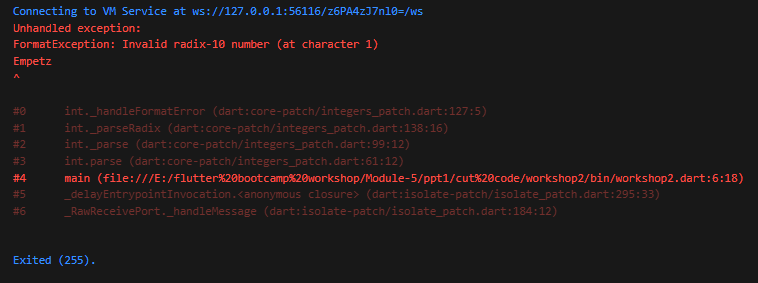
When you complete this workshop you will learn the following:

* void main(List<String> args) { ... }: Defines the entry point of the Dart program, where main is a function that takes a list of strings (args) as input.
* var x = 'Empetz';: Declares a variable x and assigns it the string 'Empetz'.
* try { ... }: Starts a try block, indicating that the code inside it might throw an exception.
* int xy = int.parse(x);: Attempts to parse the string x into an integer using the int.parse() function. Since 'Empetz' cannot be parsed into an integer, it will throw a FormatException.
* print(xy);: Prints the parsed integer value. This line will not be executed if an exception occurs in the previous line.
* } on FormatException catch (e) { ... }: Specifies that if a FormatException occurs within the try block, it should be caught, and the code inside this catch block should be executed. In this case, it simply prints the caught FormatException.

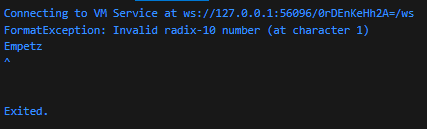
So, when you run this code, it will attempt to parse the string 'Empetz' into an integer. Since this string cannot be parsed as an integer, it will catch the resulting FormatException and print it out, allowing the program to handle the error gracefully.

# **Screenshots**

## **Before implementation (without Exception )**



## **After implementation (With Exception )**



# **How to submit your workshop**

Push your project back to the same git branch using command:

<command name>

# **Happy Coding!**