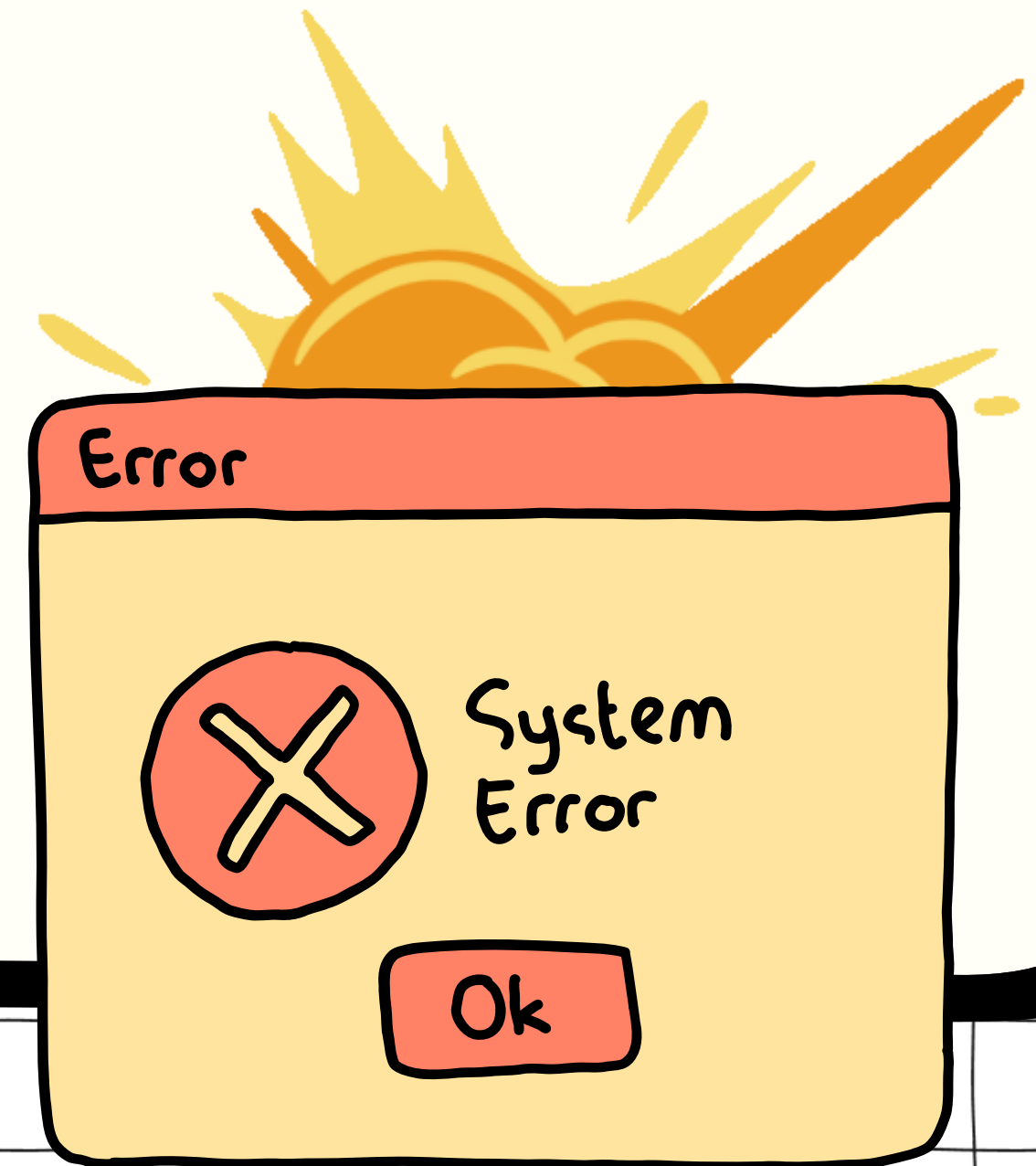


Exception Handling

C# Exception Handling



Contents

01

Introduction

02

Anatomy of C#
Exceptions

03

Try Catch Finally

04

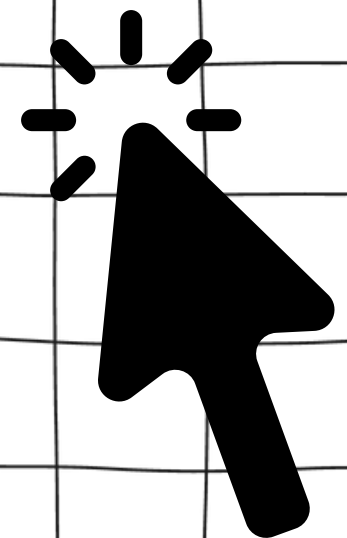
Code

05

Custom
Exception

06

Code





Introduction

Exceptions are a type of error that occurs during the execution of an application

Typically, problems that are not anticipated result in errors, whereas the application's code intentionally foresees exceptions due to various reasons.

Applications use exception handling logic to explicitly handle the exceptions when they happen. Exceptions can occur for a wide variety of reasons.

Anatomy of C# Exceptions



EXCEPTIONS ALLOW AN APPLICATION TO TRANSFER CONTROL FROM ONE PART OF THE CODE TO ANOTHER.

WHEN AN EXCEPTION OCCURS, IT DISRUPTS THE ONGOING CODE FLOW AND RETURNS CONTROL TO A PARENT TRY-CATCH BLOCK



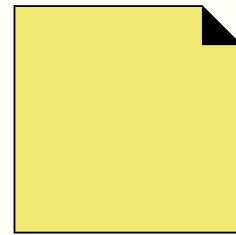
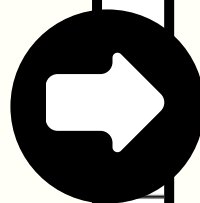
C# Exception Handling

...

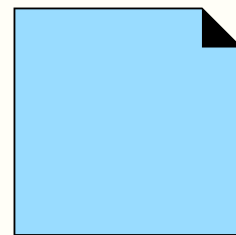
01 TRY

02 Catch

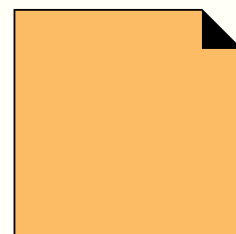
03 Finally



try block encloses a section of code.
When code throws an exception within this block, the corresponding catch handles the exception



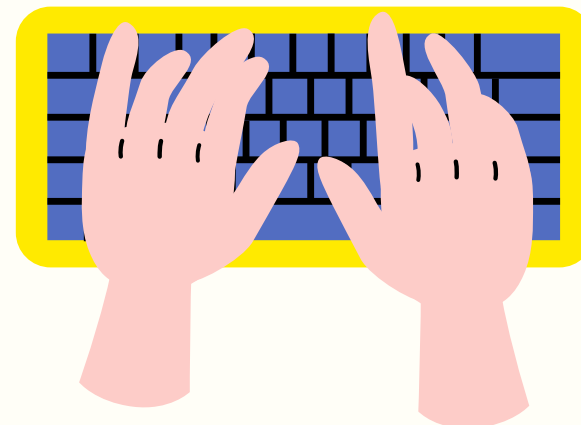
When an exception happens, the code within the Catch block executes. This is where you are able to handle the exception, log it, or ignore it



The finally block enables the execution of specific code irrespective of exception.

...

**LETS DO
SOME CODE ?**



YES

YES



Custom Exception

THROW KEYWORD



**The throw keyword crafts a
fresh exception,
ascending to a try-catch-finally
block**



**LETS DO
SOME CODE ?**



YES

YES

Thank you



/fadyehabamer

