Chapter 11

Exception Handling

Division

Description

The purpose of this application is to create an application which prompts the user numerator and denominator values then show the result of the division. The application will use exception handling to verify the user has entered valid input and will continue to prompt the user for input as long as the value they enter is invalid. Once the user has entered valid numerator and denominator values, the result is shown and the application exits.

Watch a demonstration of the application:  
  
<https://youtu.be/i90qxhqbj2M>

General Requirements

Your application must follow these general requirements

* All classes must be in a package following these rules:
  + The package name is your last name plus the first letter of your first name. For example, if you name is Rita Red, her package is “red.r”
  + Package names are all lowercase letters
* All *class* names must start with an UPPERCASE letter then camel-cased after that.
* All *property* names must start with a lowercase letter then came-cased after that.
* All *method* names must start with a lowercase letter then came-cased after that.
* Output must match the examples. Watch out for spaces and punctuation.

GradinG

***NOTE***: If the code does not compile, it’s an automatic 0!

|  |  |
| --- | --- |
| The package name is your last name plus the first letter of your first name. For example, if you name is Rita Red, her package is “red.r” | 5 |
| Print welcome message | 5 |
| Prompt user for numerator | 5 |
| Error check numerator | 5 |
| Prompt user for denominator | 5 |
| Error check denominator | 5 |
| Print result | 5 |
| Print good bye message | 5 |
| **Total:** | **40** |