Handling Double Input Exceptions

After reading the blog on Errors and Exceptions, write a C# program that reads and handles double input from the user. Use a *try-catch* block to handle any exceptions that may occur when converting the user's input to double values.

Specifically, handle *FormatException* and *OverflowException* exceptions. If an exception is caught, display an appropriate error message and ask the user to re-enter the double. Repeat this process until the user enters a valid value.

Requirements:

- Use a *try-catch* block to handle *FormatException* and *OverflowException*.
- Display specific error messages for each type of exception.
- Provide a "Try again" option after displaying an error message.
- Use a loop to allow the user to retry.
- Include a *finally* block to ensure that a message is displayed regardless of whether an exception occurs or not.

Example Output:

```
Enter a double: 12.34
You entered: 12.34
This code always gets executed, regardless of exceptions.
Exiting the Program

Enter a double: abc
Format Exception: The input string ‘abc’ was not in the correct format. Please try again.
This code always gets executed, regardless of exceptions.
Enter a double:

Enter a double: 1.7976931348623157E+400
Overflow Exception: Arithmetic operation resulted in an overflow. Please try again.
This code always gets executed, regardless of exceptions.
Enter a double:
```

Are you able to provide input to generate an OverflowException? Please check the documentation of Convert.ToDouble() here.