```python
anumber = int(input("Please enter an integer "))
Please enter an integer -23
print(math.sqrt(anumber))
Traceback (most recent call last):
  File "<pyshell#102>", line 1, in <module>
    print(math.sqrt(anumber))
ValueError: math domain error
```

```python
try:
       print(math.sqrt(anumber))
except:
       print("Bad Value for square root")
       print("Using absolute value instead")
       print(math.sqrt(abs(anumber)))

Bad Value for square root
Using absolute value instead
4.79583152331
>>>
will catch the fact that an exception
```

```python
if anumber < 0:
   raise RuntimeError("You can't use a negative number")
else:
   print(math.sqrt(anumber))

Traceback (most recent call last):
  File "<stdin>", line 2, in <module>
RuntimeError: You can't use a negative number
```

```python
def divide(x, y):
    try:
        result = x / y
    except ZeroDivisionError:
        print("Error: Cannot divide by zero!")
    else:
        print("Result of division:", result)
    finally:
        print("This will always execute, regardless of whether an exception occurred.")

# Example usage
divide(10, 2)  # Output: Result of division: 5.0 \n This will always execute, regardless of whether an exception occurred.
divide(10, 0)  # Output: Error: Cannot divide by zero! \n This will always execute, regardless of whether an exception occurred.
```