

## Error handling:

Error handling refers to the response and recovery procedures from error conditions present in a software application. In other words, it is the process comprised of anticipation, detection and resolution of application errors, programming errors or communication errors.

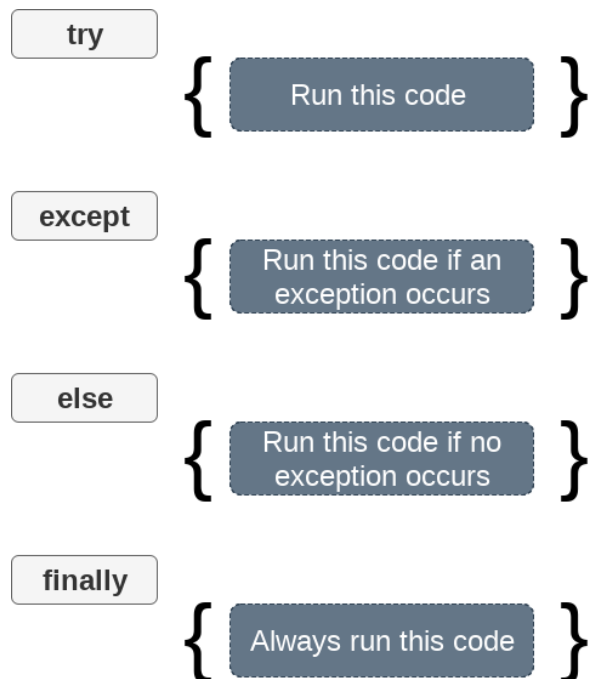
There are four main categories of errors:

1. Logical errors
2. Generated errors
3. Compile-time errors
4. Runtime errors

## Exception handling:

Exception handling is a very popular technique for incorporating fault tolerance into software systems. It allows developers to structure the redundant code that is added to deal with the exceptional conditions that may occur, separating it from the code responsible for the normal operating flow.

Example: Exception Handling in Python



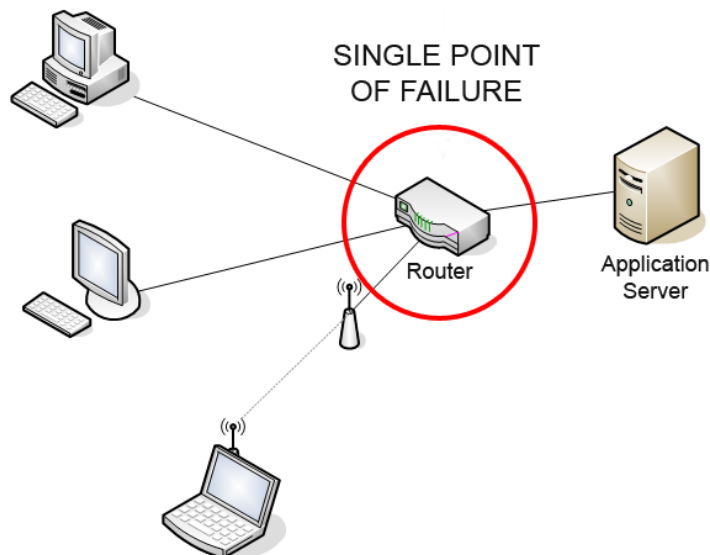
```
def divide(x, y):  
    try:  
        print(f'{x}/{y} is {x / y}')  
    except ZeroDivisionError as e:  
        print(e)
```

```
divide(10, 2)  
divide(10, 0)  
divide(10, 4)
```

## Fault Tolerance:

Fault tolerance refers to the ability of a system (computer, network, cloud cluster, etc.) to continue operating without interruption when one or more of its components fail.

The objective of creating a fault-tolerant system is to prevent disruptions arising from a **single point of failure**, ensuring the high availability and business continuity of mission-critical applications or systems.



Interesting to read:

[https://www.linkedin.com/pulse/facebook-single-point-failure-morten-rand-hendriksen/?trk=articles\\_directory](https://www.linkedin.com/pulse/facebook-single-point-failure-morten-rand-hendriksen/?trk=articles_directory)

