

Sehr gute Frage – wir klären das **Schritt für Schritt**.

Du meinst konkret diese Zeile:

```
logger.log_error(f"Account konnte nicht geladen werden: {e}")
```

Hier ist `{e}` ein **Platzhalter** im **f-String**, der automatisch die **Fehlermeldung des Exception-Objekts `e`** einfügt.



Was ist `{e}` genau?

`e` ist das **Exception-Objekt**, das du im `except`-Block so definierst:

```
except AccountNotFoundError as e:
```

Das bedeutet: Wenn ein Fehler vom Typ `AccountNotFoundError` auftritt, wird er unter dem Namen `e` gespeichert – und du kannst darauf zugreifen.

Wenn du `str(e)` verwendest (z. B. in einem f-String), bekommst du automatisch den Text, der in der Exception gespeichert wurde.



Beispiel aus deinem Projekt:

Du hast z. B. so eine Custom Exception:

```
class AccountNotFoundError(Exception):
    def __init__(self, account_id: str):
        self.account_id = account_id
        message = f"Kein Account mit der ID '{account_id}' gefunden."
        super().__init__(message)
```

Dann passiert Folgendes:

► Wenn du im Code diese Zeile wirfst:

```
raise AccountNotFoundError("abc123")
```

... und diesen Fehler später so behandelst:

```
try:
    repository.get_account("abc123")
except AccountNotFoundError as e:
    logger.log_error(f"Account konnte nicht geladen werden: {e}")
```

► Dann ergibt `{e}` diesen Text:

Account konnte nicht geladen werden: Kein Account mit der ID 'abc123' gefunden.



Wichtig zu wissen:

- Du musst **keine eigene Methode definieren**, um `{e}` verwenden zu können.
 - Die Exception selbst liefert beim Aufruf von `str(e)` automatisch die Message mit, die du im Konstruktor über `super().__init__(message)` übergeben hast.
-



Fazit für Anfänger:

`{e}` ist das **Exception-Objekt**, und wenn du es in einem f-String verwendest, wird seine **Fehlermeldung als Text** eingefügt.

Die Meldung kommt aus dem `__init__` der Klasse → also aus `super().__init__(message)`.

Wenn du möchtest, kann ich dir einmal zeigen, wie du es **schrittweise testest**, z. B. mit einer Testfunktion, in der du absichtlich einen Fehler erzeugst. Sag einfach Bescheid.