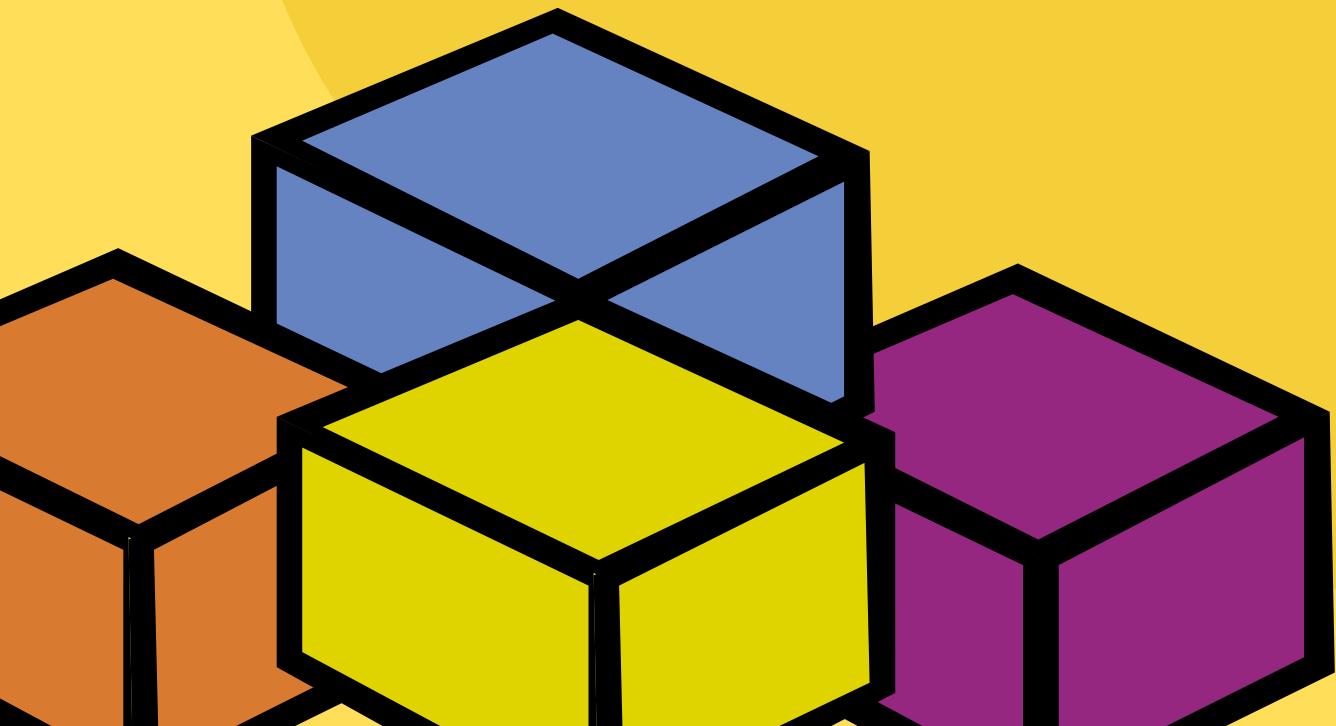
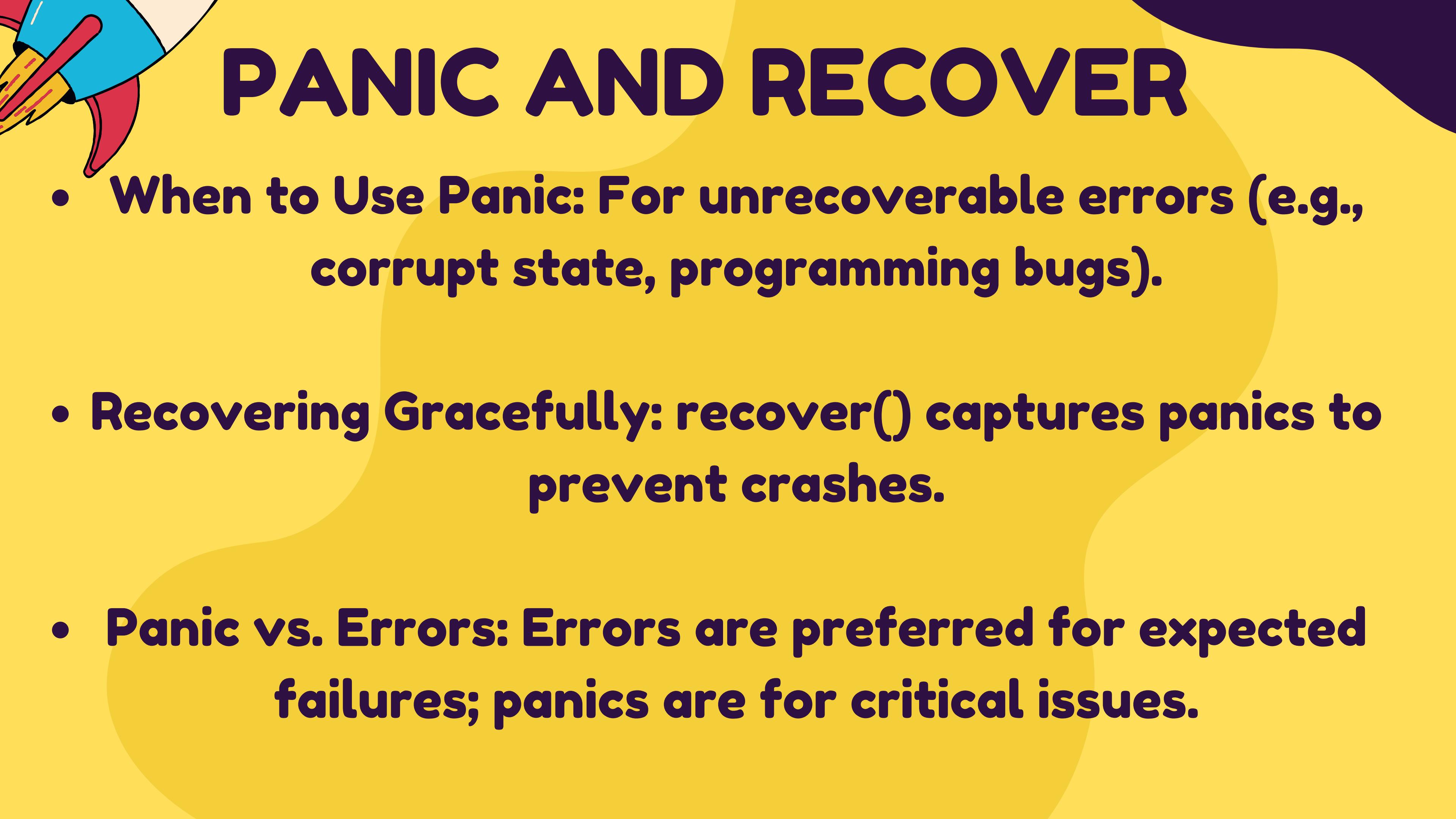


ERROR HANDLING AND TESTING



ERROR HANDLING IDIOMS

- **Returning Errors: Functions return (result, error) to indicate failures.**
- **Wrapping Errors: Use `fmt.Errorf("%w", err)` or `errors.New("message")` for context.**
- **Stack Traces & Unwrapping: The `errors` package provides `errors.Unwrap()` for nested errors.**



PANIC AND RECOVER

- **When to Use Panic:** For unrecoverable errors (e.g., corrupt state, programming bugs).
- **Recovering Gracefully:** `recover()` captures panics to prevent crashes.
- **Panic vs. Errors:** Errors are preferred for expected failures; panics are for critical issues.

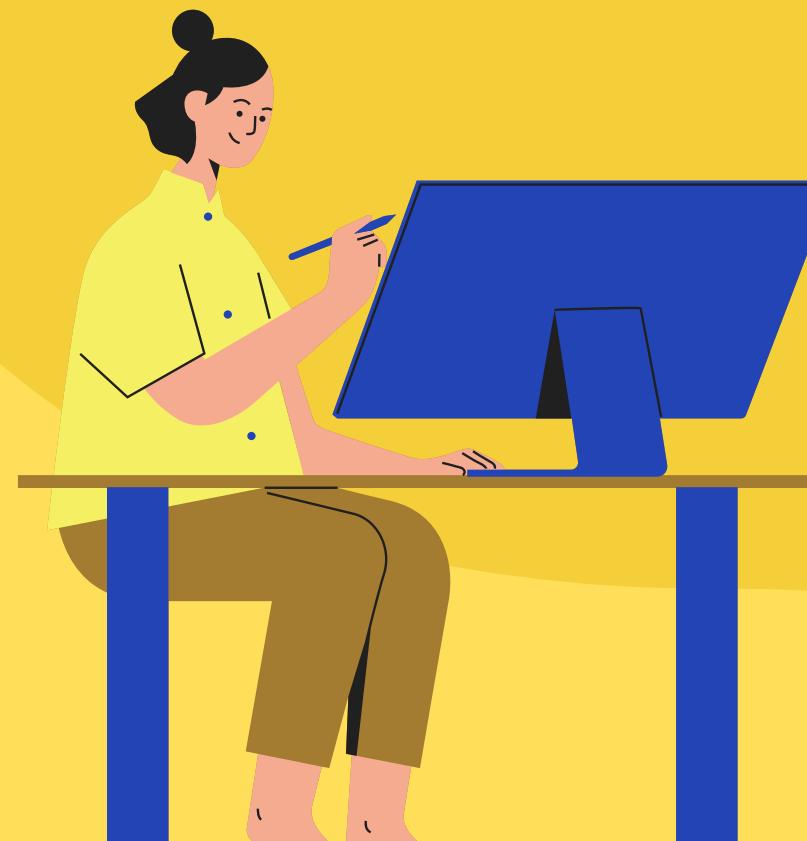
UNIT TESTING WITH TESTING PACKAGE

- TEST ORGANIZATION: FUNCTIONS MUST BE NAMED `TESTXXXX(T *TESTING.T)`.
- TABLE-DRIVEN TESTS: IMPROVE READABILITY BY USING STRUCTURED INPUT-OUTPUT CASES.



BENCHMARKING AND PROFILING

- **Writing Benchmarks:** Use `BenchmarkXxx(b *testing.B)` to measure performance.
- **Profiling with pprof:** Analyze CPU & memory usage for optimization.



THANK YOU

