

3. KISS - Keep It Simple Stupid.

Design systems as simple as possible, but no simpler.

- Simplicity over cleverness
- Reducibility over complexity.
- clarity over over-engineering -

KISS does not mean:

- Writing fewer lines at all costs.
- Avoiding design patterns.
- Avoiding abstraction completely.

KISS does mean:

- choosing the simplest solution that works.
- Avoiding unnecessary complexity.
- Solving today's problem cleanly.

Wrong Design

interface calculatorStrategy

```
int calculate (int a, int b);  
}  
class Abstract implements calculatorStrategy  
{  
    public int calculate (int a, int b)  
    {  
        return a + b;  
    }  
}
```

Eg. • use a screwdriver for screws.

- Don't bring power drill for every task.

correct design

```
int result = a + b;
```

DRY \rightarrow Avoid duplication.

KISS \rightarrow Avoid unnecessary complexity.

Sometimes slight duplication (DRY violation) is better than complex abstraction (KISS).

KISS encourages writing simple, clear solutions that are easy to understand and maintain.