|  |  |
| --- | --- |
| OOPs Design Principles | |
| KISS (Keep It Simple & Stupid) | |
| This principle has been a key, and a huge success in my years of software engineering. A common problem among software engineers and developers today is that they tend to over complicate problems.  Typically when a developer is faced with a problem, they break it down into smaller pieces that they think they understand and then try to implement the solution in code. I would say 8 or 9 out of 10 developers make the mistake that they don't break down the problem into small enough or understandable enough pieces. This results in very complex implementations of even the most simple problems, another side effect is spaghetti code, something we thought only BASIC would do with its goto statements, but in Java this results in classes with 500-1000 lines of code, methods that each have several hundreds of lines.  This code clutter is a result of the developer realizing exception cases to his original solution while he is typing in code. These exception cases would have solved if the developer had broken down the problem further. | |
| Benefit of KISS | * Solve more problems, faster. * Produce code to solve complex problems in fewer lines of code * Produce higher quality code * Build larger systems, easier to maintain * More flexible, easier to extend, modify or refactor when new requirements arrive * Achieve more than you ever imagined * Work in large development groups and large projects since all the code is stupid simple |
| How to apply the KISS principle to my work ? | |
|  | |