**Agile Values**

Agile and all of its methodologies work against 4 core values in software development.

* ***Individuals and Interactions*** over *Processes and Tools* - ensure that we value the people building a product over standard processes and tools as they are the ones who need to communicate and adapt to changes within a project.
* ***Working Software*** over *Comprehensive Documentation* - extensive documentation takes time away from software development, keep documentation streamlined and simple in order to develop software at the rapid rate needed.
* ***Customer Collaboration*** over *Contract Negotiation* - work with the customer to make the most valuable product possible rather than creating the bare minimum for a specification.
* ***Responding to Change*** over *Following a Plan* - allow customers to request changes, and embrace those changes within software development

**Note**: whilst the values on the left are more value than the right, the right-sided values are still important!

**Agile Principles**

Agile also works with 12 supporting principles in conjunction with the 4 aforementioned values.

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity - the art of maximising the amount of work *not* done is essential.
11. The best architectures, requirements, and designs emerge from self-organising teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly.