Agile software development

* 1. What key features distinguish agile development practices from traditional practices?
     1. **Agile Development- development of the application in functional components, rather then steps.**
  2. What does it mean to:
     1. be “adaptive” rather than “predictive”?
        1. **It means that your focused on adapting to changes rather then trying to plan out the risks and address them.**
     2. favor code to documentation?
        1. **According to Steve Rakitin, he says favoring documentation is a waste of time because real programmers don't write documentation**
  3. [eXtreme Programing (XP)](javascript:textise('https://en.wikipedia.org/wiki/Extreme_programming')) — focus on the introduction only here:
     1. Practices: pair programming, unit testing
        1. **Pair Programming: Code reviews being taken to the extreme.**
        2. **Unit Testing: Testing code as it is developed.**
     2. What is so “extreme” about XP?
        1. **You often will release multiple iterations of code to clients, as they change their requirements.**
  4. [Scrum](javascript:textise('https://en.wikipedia.org/wiki/Scrum_(software_development)')) — focus on the “Workflow” and “burn-down chart” sections:
     1. The meanings of these Scrum terms: sprint, daily scrum, review/retrospective, backlog and burn-down chart
        1. **Sprint: basic unit of development in a scrum.**
        2. **Daily Scrum: every day expectations are met/set for what is to be accomplished.**
        3. **Review/Retrospective: What was accomplished during the Sprint.**
        4. **Backlog: the list of requirements for a product.**
        5. **Burn-Down Chart: a chart that shows remaining time left, and how much work left is there.**
     2. The interpretation of the two lines on a burn-down chart.
        1. **Ideal Burndown- The ideal rate at which work is completed to finish the final product.**
        2. **Completed Tasks- the total amount of products finished.**
        3. **Remaining Effort: a graph of how much effort it will take at various stages of work completion.**
        4. **Reaming Tasks- the total amount of tasks remaining.**
  5. [KanBan](javascript:textise('https://en.wikipedia.org/wiki/Kanban')) — focus on the introduction only here:
     1. How does this lean manufacturing process relate to software development?

**By moving process from starting to done, it reduces the total tasks remaining to do and helps shorten the overall process.**