Scrum employs an iterative, incremental approach to optimize predictability and control risk.[[1]](#footnote-1)

Scrum Teams deliver products iteratively and incrementally, maximizing opportunities for feedback. Incremental deliveries of “Done” product ensure a potentially useful version of working product is always available.[[2]](#footnote-2)

**Product Increment**

The Increment is the sum of all the Product Backlog items completed during a Sprint and the value of the increments of all previous Sprints. At the end of a Sprint, the new Increment must be “Done,” which means it must be in useable condition and meet the Scrum Team’s definition of “Done.” It must be in useable condition regardless of whether the Product Owner decides to actually release it.[[3]](#footnote-3)

**Value Driven Development**

One anchor of agile development is incremental development. Many people, even understanding the idea of incremental development, can’t see how to break work into small, value-centered work items, and track their progress. It is easy to lose sight of the cost of rework in incremental-iterative development.[[4]](#footnote-4)

**Assessment: Product Increment**

* Definition of Product Increment: Describe that, at the end of each Sprint, the Product Owner should have the opportunity to realize value from the investment put in to date as an increment of functionality perceivable to the final user of the Product. (i.e., they could begin the deployment process for the work that has been done this Sprint, if they choose).[[5]](#footnote-5)
* Understand that Scrum leverages iterative-incremental software development principles. The software system is usually implemented in form of vertical slices. Each increment extends the previous ones. Each slice should provide value to a user or customer.[[6]](#footnote-6)
* Incremental development & staged delivery[[7]](#footnote-7):
  + What they are, their value, the difference to effort-based development.
  + Understand the basics and value of incremental development, how it differs from effort- or task-based management.
  + Understand and develop techniques for breaking problems and assignments into small value-based pieces & tracking their progress
* Importance of retaining design quality when doing incremental/iterative[[8]](#footnote-8)
  + Understand why agile developers need to keep an eye on design quality even while–or especially when–working incrementally and iteratively

1. Scrum Guide July 2016, [www.scrumguides.org](http://www.scrumguides.org) [↑](#footnote-ref-1)
2. SC [↑](#footnote-ref-2)
3. SC [↑](#footnote-ref-3)
4. Agile Fundamentals Track, www.icagile.com [↑](#footnote-ref-4)
5. CSM Learning Objectives, [www.scrumalliance.org](http://www.scrumalliance.org) [↑](#footnote-ref-5)
6. CSPO Learning Objectives, [www.scrumalliance.org](http://www.scrumalliance.org) [↑](#footnote-ref-6)
7. ICAgile [↑](#footnote-ref-7)
8. ICAgile [↑](#footnote-ref-8)