Dynamic Testing ensures the application functions correctly under different conditions and is carried out by executing actual code or the final software or hardware product. The different types of testing involve unit, integration, system, and user testing. The greatest advantage of dynamic testing is identification of runtime errors and performance issues, validation of actual functionality, and help ensure a smooth user experience.   
Static Testing identifies defects early on by reviewing documentation, code, and design is carried out against work products without actually executing a code. Within static testing, there are multiple techniques that are used such as:

* Code review
* Walkthrough
* Inspections
* Static analysis tools

Static testing also has advantages such as detecting errors before execution, reducing costs, improving code quality and security faster, and requires fewer resources.

The importance of using both Static and Dynamic Testing is that they complement each other in identifying issues throughout the SDLC by using them in order- static (before execution) and dynamic (during execution). This step allows for early detection vs Real- World Behavior. Static testing finds errors early while dynamic reveals issues in actual execution. With both, you can ensure security and performance. In this situation, static prevents vulnerabilities, dynamic testing check responsiveness under stress. Another complement from both will have comprehensive coverage, and well-rounded validation of the software.

Resources:  
[ProQuest Ebook Central - Reader](https://ebookcentral.proquest.com/lib/snhu-ebooks/reader.action?c=RVBVQg&docID=5837074&ppg=78) Chapter 3, pg 75-77