**CaseStudy : HelloSBT**

As the video progress, we’ll mark the status of action items mentioned below to “done” accordingly.

|  |  |  |
| --- | --- | --- |
|  | **Action to be taken** | **Status** |
|  |  |  |
| 1. | Check of mandatory software installation |  |
|  | Scala Installation |  |
|  | SBT Installation |  |
|  | Check SBT\_HOME environment entry |  |
|  | Check .sbt and .Ivy2 directory in C:\Users\<USER\_HOME> |  |
|  |  |  |
| 2. | Integration of Subeclipse plugin (It is used to make eclipse compatible project and its one time job only.) |  |
|  | 1. Create global.sbt in "C:\Users\<USER\_HOME>\.sbt\0.13\plugins" |  |
|  |  |  |
| 3. | Directory structure creation for HelloSBT project |  |
|  |  |  |
| 4. | Create build.sbt file and put it in the root directory of the project |  |
|  |  |  |
| 5. | Make Eclipse IDE compatible project by executing ‘eclipse’ command on SBT shell |  |
|  |  |  |
| 6. | Import the eclipse compatible project on Scala eclipse IDE |  |
|  |  |  |
| 7. | Creation of Java, Scala source code and test cases in Scala eclipse IDE |  |
|  |  |  |
| 8. | Understanding SBT basic commands e.g. clean, compile, test, package, publishLocal, publishM2 commands |  |
|  |  |  |
| 9. | Check the repository if project artifact published to Maven and Ivy repository |  |
|  |  |  |
| 10. | End of Case Study |  |