

Questão 01:

- *Validate* – Verifica se o projeto está correto;
- *Compile* – Compila;
- *Test* - Testa os códigos de testes;
- *Package* - Empacota o projeto;
- *Verify* - Verifica a integridade do pacote;
- *Install* - Instala o pacote no repositório local;
- *Deploy* - Atualiza no repositório *remote*, disponibilizando o componente para todos.

mvn package:

- Ele empacota o código compilado no formato de distribuição;
- O ciclo *default* é executado, ou seja, as fases listadas a cima (*validate*, *compile* e outros).

Questão 02:

- **<groupId>** - Identificador da organização à qual o projeto pertence;
- **<artifactId>** - É o nome do projeto;
- **<version>** - É a versão do código do projeto;
- **<mainClass>** - Serve para indicar a classe principal;
- **<finalName>** - Este é o nome final do projeto quando ele é construído;
- **<descriptorRef>** - Diz ao *Maven* que tipo de montagem esta configuração deve construir.

Questão 03:

Maven utiliza um estilo de gerencia de dependência semelhante ao *npm* onde se cria um documento que será responsável por armazenar as informações das dependências e que posteriormente são usadas na construção do projeto, porém suas semelhanças não vão muito além, suas maiores diferenças se devem ao fato de:

- *Maven* trabalha com um repositório local adicional (*cache*): O que tira a necessidade de buscar novamente a mesma dependência para projetos diferentes;
- As dependências de uma construção de projeto são baixadas em lugares diferentes;
- O processo de construção no *maven* envolve mais comumente a edição do *pom.xml*, já no *npm*, envolve escrever código ou configurar ferramentas de compilação complementares como *gulp*, *webpack* e semelhantes.