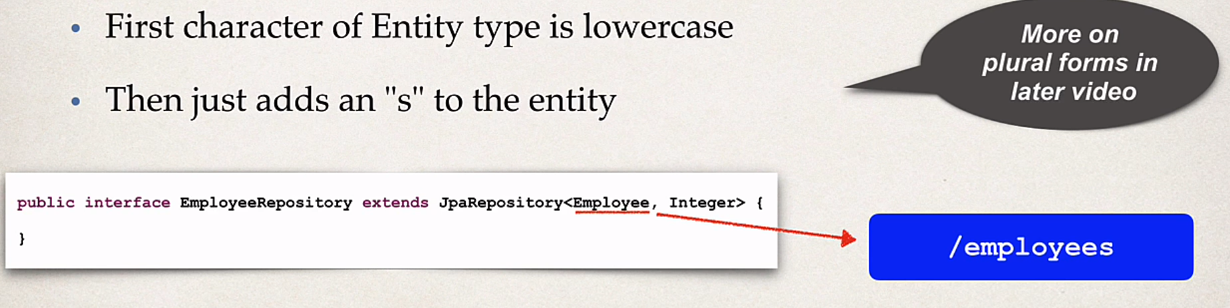
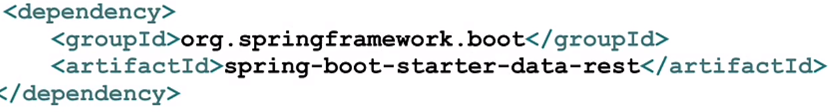
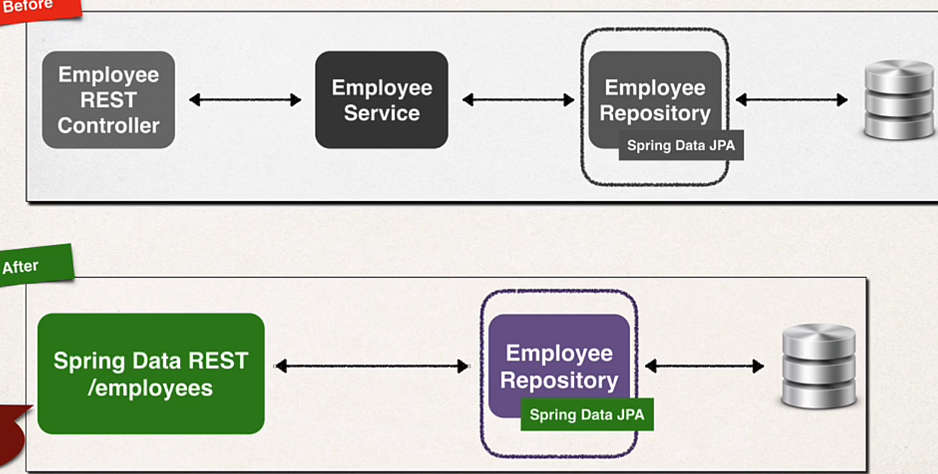
Spring data REST  
  
Abbiamo visto i benefici nell’utilizzare spring-data JPA tramite l’estensione della classe JpaRepository, ma non abbiamo ancora risolto per quanto riguarda i servizi REST che vengono esposti dai controller su determinati end-point, in quanto al momento siamo costretti per ogni Entità a riscrivere i metodi che espongono le operazioni CRUD.   
  
Al tal proposito vi viene in soccorso spring-data-rest, questa è una parte del framework spring **che semplifica la creazione di servizi RESTful** per le entità gestite da **spring data JPA**.  
Consente di esporre automaticamente le OPERAZIONI CRUD di una repository come endpoint RESTful senza dover scrivere codice per gestire esplicitamente le richieste HTTP.  
  
spring data rest identifica automaticamente i repository JPA preseneti nell’applicazione e crea endpoint restFul per le entità associate.  
  
  
  
Come si implementa spring-data-rest → aggiungendo la dipendenza spring-data-rest al file POM  
  
   
  
**Ricordiamo** che non è necessaria nessuna codifica, in quanto spring scansionando il progetto sarà capace di riconoscere automaticamente le repository e si occuperà di rendere disponibili i servizi RESTful i quali possono essere considerati come end-point ossia indirizzi url accessibili con richieste HTTP,   
e appunto si otterrano automaticamente in base ai vari tipi di entità presenti nella jpaRepository  
  
Per poter utilizzare spring-data-rest abbiamo bisogno:

* il tipo di entità,
* JPARepository
* dipendenza aggiunta al file POM

Ottenendo questo tipo di **ARCHITETTURA**  
prima avevamo la classe dao(repository) che comunicava con i service, e i service con i controller

  
  
ora utilizzando Spring data Jpa e spring data rest otteniamo la stessa cosa  
  
Inoltre è possibile personalizzare le API REST QueryDomainSpecificLanguage o DSL