**Hands-On 4: Difference between JPA, Hibernate, and Spring Data JPA**  
  
Java Persistence API , Hibernate and Spring Data JPA are the technologies used for ORM(Object -Relation Mapping) in Java Application.  
  
**Difference:  
JPA:** A Specification for relational data in Java.  
 Its just a guidelines,Do not provide any implementation.  
 does not Manages Transactions.  
  
**Hibernate**: An Implementation of JPA .  
 Concrete ORM Framework .  
 It can Manage Transactions.  
  
**Spring Data JPA:** An Abstraction Layer top of JPA providers like hibernate .  
 Uses existing JPA implementations.  
 Provides ready made CRUD operations.   
 It can Manage Transactions.  
  
 **Code Snippets:**  
**Hibernate :**  
 public Integer addEmployee(Employee employee){

      Session session = factory.openSession();

      Transaction tx = null;

      Integer employeeID = null;

      try {

         tx = session.beginTransaction();

         employeeID = (Integer) session.save(employee);

         tx.commit();

      } catch (HibernateException e) {

         if (tx != null) tx.rollback();

         e.printStackTrace();

      } finally {

         session.close();

      }

      return employeeID;

   }

**Spring Data JPA:**  
// EmployeeRepository.java

public interface EmployeeRepository extends JpaRepository<Employee, Integer> {

}

// EmployeeService.java

@Service

public class EmployeeService {

@Autowired

private EmployeeRepository employeeRepository;

@Transactional

public void addEmployee(Employee employee) {

employeeRepository.save(employee);

}

}  
  
  
**Summary:**  
 JPA defines what ORM should do .  
 Hibernate implements the JPA standards and provides extra features.  
 Spring Data JPA uses Hibernate under the hood and removes the boilerplate code by providing ready-to -use repository interface.