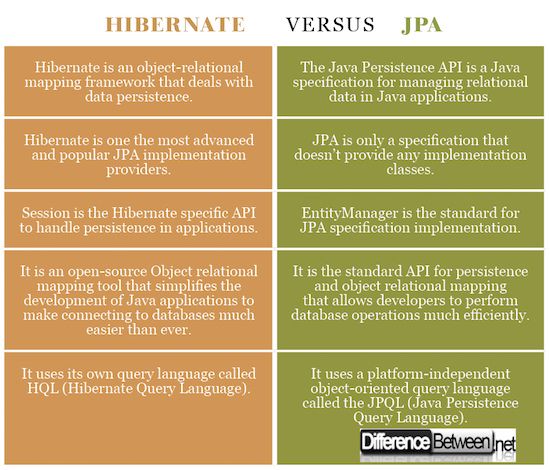
***Hibernate***

1. ***Какво е.***  
   Hibernate е рамка, която може да се използва за обектно-релационно картографиране, предназначено за програмен език Java. По-конкретно, това е ORM (обектно-релационно картографиране) библиотека, която може да се използва за картографиране на обектно-релационен модел в конвенционален релационен модел. С прости думи, той създава картографиране между Java класове и таблици в релационни бази данни, също и между Java и SQL типове данни. Хибернацията може да се използва и за заявки и извличане на данни чрез генериране на SQL повиквания. Следователно програмистът е освободен от ръчното боравене с набори от резултати и преобразуване на обекти. Hibernate се пуска като безплатна рамка с отворен код, разпространявана под лиценз GNU. Реализация за JPA API е предоставена в Hibernate 3.2 и по-нови версии. Гевин Кинг е основателят на Hibernate.

***2.Разлика между хибер и jpa***

JPA е рамка за управление на релационни данни в Java приложения, докато Hibernate е специфично изпълнение на JPA (така че в идеалния случай JPA и Hibernate не могат да бъдат директно сравнени). С други думи, Hibernate е една от най-популярните рамки, която прилага JPA. Hibernate прилага JPA чрез Hibernate Annotation и EntityManager библиотеки, които са внедрени върху Hibernate Core библиотеки. И EntityManager, и Annotations следват жизнения цикъл на Hibernate. Най-новата версия на JPA (JPA 2.0) се поддържа изцяло от Hibernate 3.5. JPA има предимството да има интерфейс, който е стандартизиран, така че общността на разработчиците ще бъде по-запозната с него от Hibernate. От друга страна, родните API на Hibernate могат да се считат за по-мощни, тъй като неговите функции са супер набор от тези на JPA  
  
Почти всички корпоративни приложения се нуждаят от някакъв начин за постоянно съхранение на данни. От жизненоважно значение за корпоративните приложения поради необходимия достъп до релационни бази данни. Постоянството е основна концепция при разработването на приложения и само по себе си е предизвикателство поради необходимостта от сериализиране на йерархично структурираните Java обекти към база данни по начин, който е едновременно бърз и ефективен. Въпреки че постоянството вече е гореща тема за дискусии в общността на Java, много разработчици дори няма да се съгласят относно обхвата на проблема. В контекста на обектно-ориентирания език, постоянството означава, че данните са все още достъпни дори след завършилия процес, който ги е създал. Има много начини за постоянно съхранение на данни. Hibernate е един такъв проект, който има за цел да осигури цялостно решение на проблема с постоянството в Java, докато JPA е спецификация за управление на постоянни данни и обектно-релационни картографирания.



Хибернацията е сравнително лесна за разбиране и прилагане

Хибернация не е толкова склонна към грешки като EJB

Хибернацията от друга страна е инструмент за ORM, който няма стръмна крива на обучение, която е обща с EJB и не е толкова склонна към грешки като EJB. Хибернацията е предпочитана поради своята гъвкавост, което позволява използването на Servlets и JSP. Hibernate е изграден в съответствие със спецификацията J2EE, която го прави приложим в повечето приложения, които разработчиците ще имат. Hibernate е специален инструмент за ORM, тъй като не може да се използва само в контейнери EJB, но също така е приложим и в самостоятелни контейнери, и в уеб контейнери. Това е основната разлика, която се откроява между EJB и Hibernate. Следователно развитието с хибернация е много по-лесно и по-малко сложно за разлика от използването на EJB.