**Q1.**\_\_\_\_\_\_\_\_\_\_\_\_   interface shoule be used when the query result type is unknown or when a query returns polymorphic results .

**1)**. **TypedQuery**

**2)**. **Query**

**3)**. EntityManager

**4)**. Non of the above

**Q2.** What is true about the following code:  
  
@OneToMany(mappedBy="department")   
 public Collection<Employee> getEmployees() {   
 return employee;   
 }

**1)**. **Department has a field called “Employee”, that has a one-to-many relationship**

**2)**. **Each “department” has one or more “Employee”**

**3)**. Each “Employee” has one or more “department”

**4)**. **None of these**

**Q3.** State whether below given statements are true or false:   
Statement 1-            JPQL is a platform-independent object-oriented query language defined as part of the Java Persistence API (JPA) specification.  
  
Statement 2 - JPQL works with relational database Table not java classes .

**1)**. **True    True**

**2)**. **True    False**

**3)**. False   True

**4)**. False    False

**Q4.**Which of the following entitymanager method is used to locate single entity only based on primary key value .

**1)**. search()

**2)**. **locate()**

**3)**. flush()

**4)**. **find()**

**Q5.**Which of the below given strategies JAP allows for hierarchical classes to be mapped with tables?

**1)**. **InheritanceType.SINGLE\_TABLE**

**2)**. InheritanceType.  
TABLE\_PER\_CLASS

**3)**. InheritanceType.JOINED

**4)**. **All of the above**

**Q6.** @Entity  
public class Employee ….. {  
 @Id  
 private int empId;  
 private String name;  
 @OneToOne  
 private Address address;  
}  
  
@Entity  
public class Address ….. {  
 @Id  
 private int addressId;  
 private String street;  
 private String city;  
 private String state;  
 private String zipcode;}  
Which of the given statement is true about above code?

**1)**. **It represents unidirectional one-to-one relationship**

**2)**. It represents bidirectional one-to-one relationship

**3)**. It represents bidirectional one-to-many relationship

**4)**. **None of these**

**Q7.** Which of the given below are valid cascade types?

**1)**. **Detach**

**2)**. **Merge**

**3)**. Insert

**4)**. **Remove**

**5)**. **All**

**Q8.** How can we represent the Many to Many Entity relationship in JPA?

**1)**. **javax.persistence.ManyToMany**

**2)**. javax.persistence.Entity

**3)**. javax.persistence.Table

**4)**. javax.hibernate.ManyToMany

**Q9.**Query interface shoule be used in which of the following case as per JPA 2 recommendations?

**1)**. When specific query result type is expected

**2)**. **When query result type is unknown**

**3)**. When query result contains more than one object

**4)**. **When query result contains only one object**

**Q10.** Which of the given syntax is correct to use Named Queries?

**1)**. **@Entity  
@Table(name = "books")  
@NamedQueries(  
@NamedQuery(name = "getAllBooks", query = "SELECT book FROM Book book"))  
public class Book implements Serializable {  ………….. }**

**2)**. @Entity  
@Table(name = "books")  
@NamedQueries(  
name = "getAllBooks", query = "SELECT book FROM Book book")  
public class Book implements Serializable {  ………….. }

**3)**. @Entity  
@Table(name = "books")  
@NamedQueries(  
@NamedQuery(name = "getAllBooks", query = "SELECT book FROM Book book"))  
public class Book {  ………….. }

**4)**. @Entity  
@Table(name = "books")  
@NamedQuery(  
@NamedQueries(name = "getAllBooks", query = "SELECT book FROM Book book"))  
public class Book implements Serializable {  ………….. }