

```

@Service
public class AppService {
    @Autowired
    private BookRepo bookRepo;
}

@Service
public class AdminService {
    @Autowired
    private BookRepo bookRepo;
}

```

```

@Repository
@Scope("prototype") ✓
//@Profile("prod")
public class BookRepoDbImpl implements BookRepo{
    ...
}

```

```

@Service
public class AppService {
    @Autowired
    private BookRepo bookRepo;
}

@Service
public class AdminService {
    @Autowired
    private BookRepo bookRepo;
}

```

```

@Repository
@RequestScope ✓
public class BookRepoDbImpl implements BookRepo{
    ...
}

```

1 Request →

```

@Repository
@RequestScope ✓
public class BookRepoDbImpl implements BookRepo{
    ...
}

```

2 Request →

```

@Repository
@RequestScope ✓
public class BookRepoDbImpl implements BookRepo{
    ...
}

```

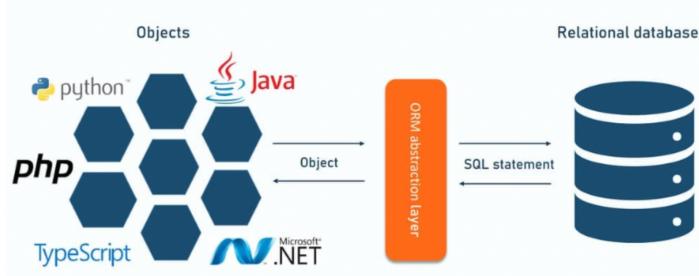
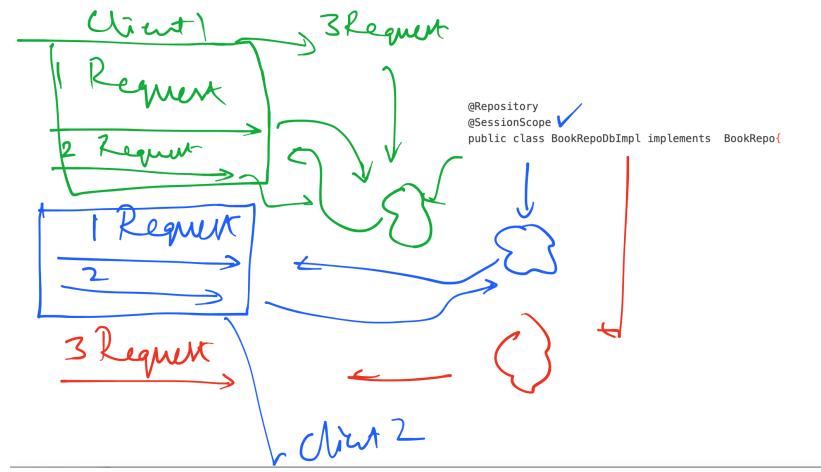
3 Request →

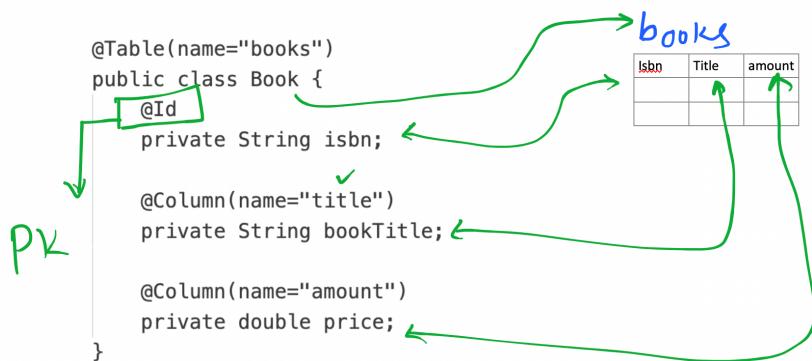
```

@Repository
@RequestScope ✓
public class BookRepoDbImpl implements BookRepo{
    ...
}

```

Session Scope:





JDBC

```
    @Override
    public void addProduct(Product product) throws DaoException {
        Connection con = null;
        try {
            String SQL = "INSERT INTO products (id, name, price, category) VALUES (?, ?, ?, ?)";
            PreparedStatement ps = con.prepareStatement(SQL);
            ps.setString(1, product.getName());
            ps.setDouble(2, product.getPrice());
            ps.setInt(3, product.getQuantity());
            ps.executeUpdate();
        } catch (SQLException e) {
            if(ex.getErrorCode() == 1062) {
                throw new DaoException("Product with given id: " + product.getId() + " already exists!");
            } else if(ex.getErrorCode() == 1054) {
                throw new DaoException("Unable to add Product!!!");
            }
        } finally {
            DBUtil.closeConnection(con);
        }
    }
}
```

BRM

```
public void addProduct(Product product)
    em.persist(product);
}
```

