```
interface EmployeeDao {
    addEmployee(Employee e);
}

public class EmployeeDaoMongoImpl implements EmployeeDao {
    addEmployee(...)
}

public class EmployeeDaoSQLImpl implements EmployeeDao {
    addemployee(...)
}

public class AppService {
    EmployeeDao empDao; // loosly coupled

    public void setDao(EmployeeDao edao) {
        this.empDao = edao;
    }

    insert(Employee e) {
        this.empDao.addEmployee(e);
    }
}
```

```xml
<beans>
    <bean id="mongo" class="pkg.EmployeeDaoMongoImpl" />
    <bean id="sql" class="pkg.EmployeeDaoSQLImpl" />
    <bean id="service" class="pkg.AppService">
        <property name="dao" ref="sql" />
    </bean>
</beans>
```
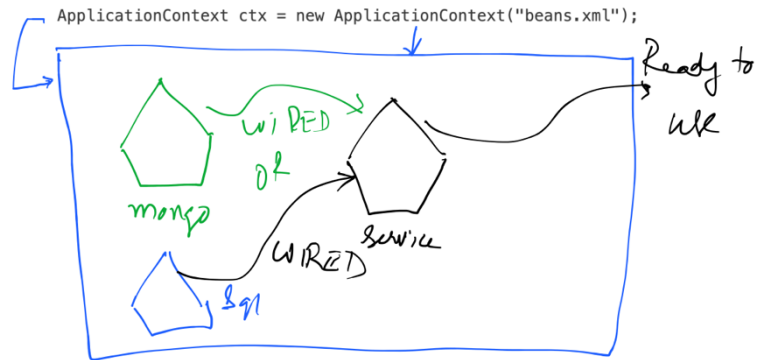
setDao(↓)   mongo

```
ApplicationContext ctx = new ApplicationContext("beans.xml");
```

Ready to use

WIRED
OR
mongo

WIRED
Service

sql

@Autowired
private EmployeeDao employeeDao;

OR

ERROR

@Repository
public class EmployeeDaoMongoImpl implements EmployeeDao {

@Repository
public class EmployeeDaoSqlImpl implements EmployeeDao {

B Service
@Autowired
@Qualifier("employeeDaoSqlImpl")
private EmployeeDao employeeDao;

@Repository
public class EmployeeDaoMongoImpl implements EmployeeDao {

@Repository
public class EmployeeDaoSqlImpl implements EmployeeDao {

A Service
@Autowired
@Qualifier("employeeDaoMongoImpl")
private EmployeeDao employeeDao; //
```

# JDBC

```java
@Override
public void addProduct(Product product) throws PersistenceException {
    String SQL = "INSERT INTO products(id, name, price, quantity) VALUES(0, ? ,? ,?)";
    Connection con = null;
    try {
        con = DBUtil.getConnection();
        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) {
        // log
        throw  new PersistenceException("unable to add product", e);
    } finally {
        DBUtil.closeConnection(con);
    }
}

@Override
public List<Product> getProducts() throws DaoException {
    String SQL = "SELECT id, name, price, quantity FROM products";
    Connection con = null;
    List<Product> products = new ArrayList<>();
    try {
        con = DBUtil.getConnection();
        Statement stmt = con.createStatement();
        ResultSet rs = stmt.executeQuery(SQL);
        while(rs.next()) {
            products.add(new Product(
                    rs.getInt("id"),
                    rs.getString("name"),
                    rs.getDouble("price"),
                    rs.getInt("quantity")
            ));
        }
    } catch (SQLException e) {
        // log
        throw  new DaoException("unable to get products", e);
    } finally {
        DBUtil.closeConnection(con);
```

ORM

```java
public void addProduct(Product product) {
    em.save(product);
}
```

```java
public List<Product> getProducts() {
    TypedQuery<Product> query = em.createQuery(Product.class);
    return em.getQueryResult();
}
```

Contacts

PersistanceContext

EntityManagerFactory

@Entity

EntityManager

Hibernate JPA Vendor

O O O O O O O ← DataSource
Pool of database Connection

Sync

RDBMS