

软件构造 实验指导书

太原科技大学
计算机学院

目 录

实验 1 软件构造的前期准备工作.....	1
实验 2 构造中的设计-设计可以工作的类.....	2
实验 3 系统结构设计及用户登录实现.....	3
实验 4 添加培训计划.....	5
实验 5 员工管理.....	8
实验 6 部门管理.....	10
实验 7 培训计划学员安排.....	13
实验 8 培训成绩管理.....	16
附件：	19
一、 Power Designer 创建 PDM（物理模型）教程.....	19
二、 WindowBuilder 离线安装过程.....	23
三、 WindowBuilder 使用.....	31
四、 DAO 设计模式.....	34
五、 Apache DBUtilits 框架.....	38

实验 1 软件构造的前期准备工作

一. 实验目的

通过实际案例使学生理解软件构造的前期准备工作的重要性，以及如何验证前期准备工作，理解迭代式和序列式软件开发的使用场景。

二. 实验内容

1. 打开《电厂职工培训管理系统》的原型系统，理解该软件的功能需求。
2. 根据该系统的功能需求，撰写一个简要的需求规格说明书，并参照教材中的需求核对表，检查需求准备工作是否完成。
3. 使用 Power Designer 工具菜单中的逆向工程，创建本系统的物理模型，结合软件的需求，对表以及其中的字段加上中文注释，并建立表之间的关系。通过此项工作加深对需求的理解。

一、将创建的好的 PDM 物理模型，导入到 MySQL 数据库中，为下一步开发做好准备。也可参考附录中的系统数据库物理模型初步设计，自己完成数据库的创建。

三. 实验过程及结果

- 1、以附件形式打印本系统的需求规格说明书
- 2、以附件形式打印系统的数据库模型
- 3、提出你对给出的数据模型的意见、建议及相应的理由。

四. 实验遇到问题及体会

实验 2 构造中的设计-设计可以工作的类

一. 实验目的

通过实验 1 对《电厂职工培训管理系统》的需求进行梳理，本实验结合系统实际需求，运用软件构造的多种启发式方法，如面向对象方法、信息隐藏、形成一致的抽象以及封装，并使用迭代、分而治之以及自上而下等多种设计实践，设计《电厂职工培训管理系统》的类结构。

通过本次实验使学生充分和理解软件技术的首要使用是管理复杂度，软件设计的特点，好的设计都是经过反复迭代，多次演化逐渐形成的。

二. 实验内容

1. 给出构造本系统的所使用的规范，包括类命名、方法命名及属性命名等方面。
2. 给出系统类结构图。
3. 根据 20/80 原则，对主要类的结构进行描述，包括每个类对外提供的接口（public method）及类之间的关系等方面。
4. 根据良好类设计的原则（良好的接口、一致抽象层次、封装等），检查设计结果是否符合要求。

三. 实验过程及结果

- 1、说明类设计的结果，以及选择此组织结构的理由，并以附件形式打印本系统的类结构图

四. 实验遇到问题及体会

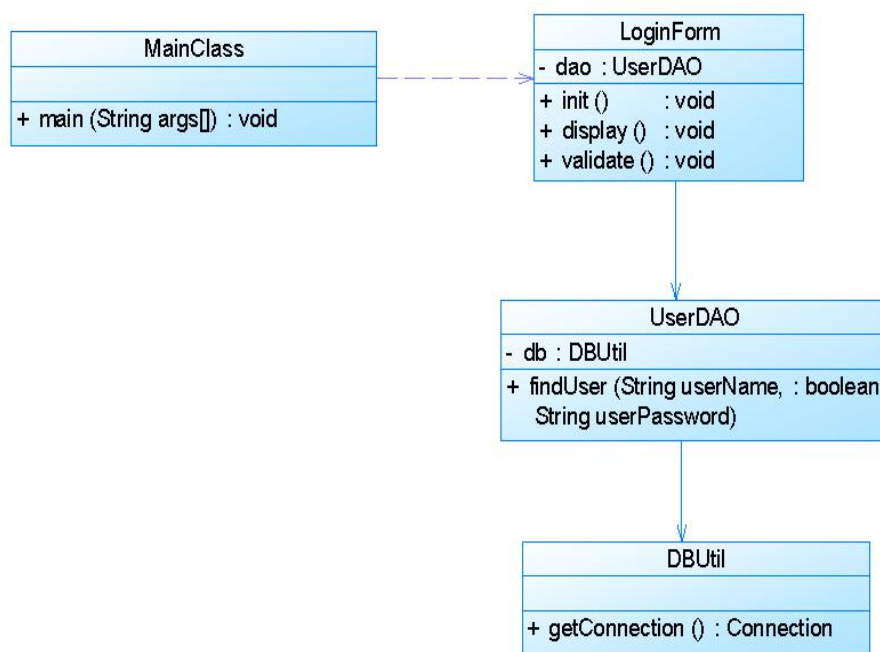
你的类设计结果是一气呵成的吗？对什么地方进行了修改，理由如何？

实验 3 系统结构设计及用户登录实现

一. 实验目的

1. 通过本次实验使学生理解掌握常规软件的层次化设计方法，通常将软件分解为视图层、业务层、持久层。其中视图层负责数据的输入和输出；业务层负责业务逻辑的实现；持久层负责数据库的增删改查等访问操作；

2. 实现用户登录功能，可按照如下类图实现该功能。LoginForm 类负责登录界面的初始化、显示、数据性验证功能的实现；UserDAO（用户数据访问对象）负责实现判断用户的合法性检验；DBUtil 类是一个工具类，负责 JDBC 连接的创建、关闭以及查询的执行等数据库操作。



二. 实验内容

1. 在 eclipse 环境下安装 windowbuilder 插件。该插件可以可视化和所见即所得方式用 swing 组件开发软件。具体安装过程参考[附件一](#)、[附件二](#)

2. 用安装好的插件开发本系统的登录界面（参考原型），并实现相应功能。

三. 实验过程及结果

1. 以图片和适当文字说明形式，写出插件安装过程。

2. 打印登录功能源代码，需要注意代码风格（注释、命名等方面）。

四. 实验遇到问题及体会

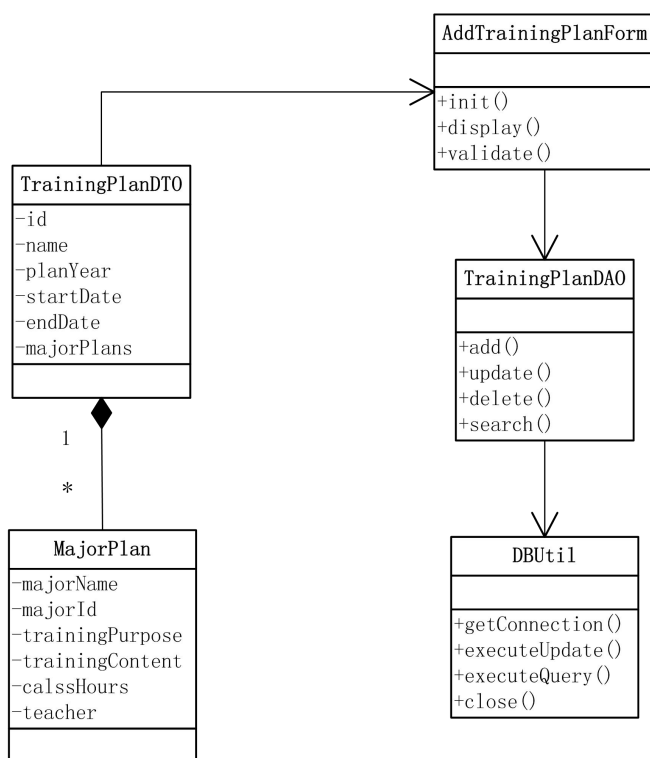
实验 4 添加培训计划

一. 实验目的

实现培训计划的增加功能，通过该功能的实现使学生理解类设计、特别是方法设计、变量命名等软件构造的理论知识。

二. 实验内容

遵循 DAO 设计模式（[参照附件三](#)），并按照如下图方式实现培训计划的添加功能。



AddTrainingPlanForm 类：表示培训计划的添加界面		
1	init	实现界面的初始化
2	display	负责界面的显示
3	Validate	验证用户输入的培训计划数据的合法性

TrainingPlanDTO 类：数据传输对象，表示一个培训计划实体		
1	id	编号
2	name	培训计划名称

3	planYear	年度
	startDate	开始时间
	endDate	结束时间
	majorPlane	各专业培训计划内容

MajorPlan 类：数据传输对象，表示一个专业的培训计划

1	majorName	专业名称
2	majorId	专业编号
3	trainingPurpose	目的
	trainingContent	内容
	classHours	课时
	Teacher	培训教师

TrainingPlanDAO 类：数据访问对象，用于执行对培训计划的增删改查的操作

1	add	添加
2	update	修改
3	delete	删除
4	Search	查询

DBUtil 类：数据库工具类，实现数据库连接的创建、关闭、SQL 语句执行操作

1	getConnection	创建连接
2	executeUpdate	执行 insert、update 语句
3	executeQuery	执行 select 语句
4	close	关闭连接

三. 实验过程及结果

- 1、按照类图以及类的说明实现功能，需要注意代码风格（注释、命名等方面）。
- 2、打印该功能的源代码，以及测试结果；

四. 实验遇到问题及体会（必须要写）

- 1、在本次实验中你应用到哪些软件构造的知识，并举例说明。
- 2、在实现过程遇到了什么问题，以及如何解决的？
- 3、讲讲本次实验的收获。

实验 5 员工管理

一. 实验目的

实现员工信息的添加、修改、删除和查询功能，通过该功能的实现使学生理解类设计、特别是方法设计、变量命名等软件构造的理论知识。

二. 实验内容

老师提供的代码已实现的功能：

1、系统 MDI 框架。MDI (Multiple Document Interface) 即多文档界面。使用 MDI 窗体时，将在一个父窗体内建立工作区，父窗体能够令一个以上的子窗体限制于其中活动及操作。；

2、搭建好了系统界面框架，包括菜单、工具栏、子窗口；

3、实现了添加员工功能界面；

要求在老师已实现功能的基础上，实现以下功能

1、员工信息的保存：要求在实现时

a) 首先要对界面中的信息的合法性进行语法验证，包括必填项是否填写、数据合法性检验（身份证、出生日期应该大于 18 岁）。若有不合法信息应给出提示；

b) 验证通过后，将输入员工信息保存到数据库中

2、删除员工：首选在员工 Table 中选择某条信息，然后点击删除按钮，因此，在点击删除按钮时，需要判断用户是否选择了员工，如果选择了，首选提示用户是否要删除该员工，用户确认后，执行删除操作，删除数据库中对应的记录。删除成功后，刷新 Table 中信息。

3、修改员工：与删除功能类似，首先从 Table 中选择一条要修改的记录，因此，在点击修改按钮时，需要判断用户是否选择了员工，如果选择了，获得当前所选择的员工的信息，接着，打开修改员工界面（该界面可以单独做，或者重用现在添加员工信息界面），与保存员工类似，点击保存按钮时要进行语法验证，通过后保存到数据库中。

4、员工查询：可以根据用户名、用户性别查询，并将查询结果显示到 Table 中。

三. 实验过程及结果

- 1、阅读理解老师提供的源代码，熟悉程序框架、功能实现过程。
- 2、按照实验要求完成相应功能
- 3、打印该功能的源代码，以及测试结果；

四. 实验遇到问题及体会（必须要写）

- 1、在本次实验中你应用到哪些软件构造的知识，并举例说明。
- 2、在实现过程遇到了什么问题，以及如何解决的？
- 3、讲讲本次实验的收获。

实验6 部门管理

一. 实验目的

实现部门信息的添加、修改、删除和查询功能，通过该功能的实现使学生理解类设计、特别是方法设计、变量命名等软件构造的理论知识。

二. 实验内容

部门管理模块的主界面如下图所示：

部门

太原第二热电厂

发电一部

发电二部

检修车间

锅炉车间

燃料车间

添加 修改 删除

共计: 10条信息

导出 打印

序号	部门名称	部门编号	部门简称	联系方式	备注
1	发电一部	001001			
2	发电一部	001001			
3	发电一部	001001			
4	发电一部	001001			
5	发电一部	001001			
6	发电一部	001001			
7	发电一部	001001			
8	发电一部	001001			
9	发电一部	001001			
10	机电科	001001			

1 2 3 4 ... >

该模块需要实现以下功能：

1、以树形结构显示部门名称：当首次显示该界面时，系统从数据库中读取部门信息，并以树形结构（JTree）显示该单位所有部门名称。

2、以列表方式显示部门信息：当首次显示该界面时，系统从数据库中读取部门信息，并以列表（JTable）显示该单位所有部门信息，包括名称、编号、联系方式等内容。

3、添加部门：点击【添加】按钮，如下图所示，弹出一个模态对话框。

a) 系统首先读取左边树形结构结点（node）所选择的部门信息，加载到界面，即默认是向该部门添加下属部门。

b) 当点击保存按钮时，系统要对界面信息进行语法验证；

c) 验证通过后根据上级部门编号生成本部门编号：若上级部门：001，当前部门：001***，其中***是根据 001 直接下属部门最大编号+1 计算，此功能可以单独方法实现；

d) 将当前信息保存到部门表中

4、修改部门

- 首先判断用户在列表中是否选择了需要修改的部门
- 若没有，提示用户选择；
- 若已选择，读取用户选择的部门信息，为修改做好准备
- 与添加部门功能类似，弹出一个模态对话框，将需要修改部门信息显示到界面中，其中上级部门以及当前部门编号为只读，不允许修改。
- 当用户点击，当点击保存按钮，要进行语法验证，通过后保存到数据表中。

5、删除部门

- 首先判断用户在列表中是否选择了需要修改的部门
- 若没有，提示用户选择；
- 若已选择，弹出一个对话框，询问用户是否确认删除该部门信息。
- 若确认删除，读取用户选择的部门编号，为删除做好准备
- 根据部门编号从数据表中删除对应记录。

三. 实验过程及结果

- 1、按照实验要求完成相应功能
- 2、打印该功能的源代码，以及测试结果；

四. 实验遇到问题及体会（必须要写）

- 1、在本次实验中你应用到哪些软件构造的知识，并举例说明。
- 2、在实现过程遇到了什么问题，以及如何解决的？
- 3、讲讲本次实验的收获。

实验7 培训计划学员安排

一. 实验目的

实现培训计划计划信息展示、学员的安排功能，通过该功能的实现使学生理解类设计、特别是方法设计、变量命名等软件构造的理论知识。

二. 实验内容

1、培训计划信息显示，如下图所示。当用户点击培训计划管理菜单时，系统按照时间顺序将当前所有培训计划信息显示到 JTable 中。此项功能需要将培训计划表与学员表关联，统计出培训人数。

序号	年度	培训计划名称	培训类型	培训专业	培训时间	培训人数	完成情况
1	2015	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	100	未培训
2	2015	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	未安排	未培训
3	2015	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	100	未培训
4	2015	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	100	未培训
5	2014	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	100	未培训
6	2014	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	100	已培训
7	2014	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	100	已培训
8	2014	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	100	已培训
9	2014	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	100	已培训
10	2014	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	100	已培训

2、培训计划所属专业信息展示：用户点击上图中的【未安排】链接，显示出如下界面，将当前培训计划对应的每个专业培训信息在列表中展示。

培训计划基本信息								
年度	2015		名称	新员工入厂培训		时间	1月10日-1月20日	
专业	培训目的		培训内容		课时	授课人	操作	
电气	电工基础知识: 1、直流电路 2、电磁与电磁感应 3、交流电路		1 掌握直流电路组成及作用。 2、了解电磁怎样产生及作用。 3 交流电怎样产生		3	吴浩	安排学员	
汽机	电工基础知识: 1、直流电路 2、电磁与电磁感应 3、交流电路		1 掌握直流电路组成及作用。 2、了解电磁怎样产生及作用。 3 交流电怎样产生		3	吴浩	安排学员	
锅炉	电工基础知识: 1、直流电路 2、电磁与电磁感应 3、交流电路		1 掌握直流电路组成及作用。 2、了解电磁怎样产生及作用。 3 交流电怎样产生		3	吴浩	安排学员	

3、学员安排：当点击上图中的【安排学员】链接时，显示如下所示界面。

a) 界面初始化：与部门管理功能类似，首先从部门表中读取所有部门信息，并以树形和列表方式显示出来。

b) 员工信息动态展示：当用户点击左边部门名称时，系统能动态将该部门下属的员工显示到右边列表中。

c) 选择学员：能够将员工列表中选择的信息，添加到已安排学员列表中。在添加时候，要对重复的员工进行判断。此功能关键点是 JTable 数据行 row 的复制。

d) 删除已选择学员：可以允许删除已选择的学员。

e) 保存功能：能够将已选择学员列表中信息添加到对应的数据表中。

学员安排 X

太原二电厂

- ▼ 发电一部
 - 电气1班
 - 汽机1班
 - 锅炉1班
- 发电二部
 - 电气车间
 - 燃料车间
 - 粉煤灰车间

部门：发电一部 电气1班

选择

全选	姓名	职务	入厂时间	岗位
<input type="checkbox"/>	张志刚	班长	1999.8	
<input type="checkbox"/>	李峰	班员	2014.9	
<input type="checkbox"/>	李峰	班员	2014.9	
<input type="checkbox"/>	李峰	班员	2014.9	
<input type="checkbox"/>	李峰	班员	2014.9	
<input type="checkbox"/>	李峰	班员	2014.9	
<input type="checkbox"/>	李峰	班员	2014.9	
<input type="checkbox"/>	李峰	班员	2014.9	
<input type="checkbox"/>	李峰	班员	2014.9	
<input type="checkbox"/>	李峰	班员	2014.9	

已安排学员 10 人

删除

保存

姓名	部门	姓名	部门
张志刚	发电一部 电气1班	张志刚	发电一部 电气1班
李峰	发电一部 电气1班	李峰	发电一部 电气1班
李峰	发电一部 电气1班	李峰	发电一部 电气1班
李峰	发电一部 电气1班	李峰	发电一部 电气1班
李峰	发电一部 电气1班	李峰	发电一部 电气1班

三. 实验过程及结果

- 1、按照实验要求完成相应功能
- 2、打印该功能的源代码，以及测试结果；

四. 实验遇到问题及体会（必须要写）

- 1、在本次实验中你应用到哪些软件构造的知识，并举例说明。
- 2、在实现过程遇到了什么问题，以及如何解决的？
- 3、讲讲本次实验的收获。

实验 8 培训成绩管理

一. 实验目的

实现学员的培训成绩的录入、修改、查询和删除功能，通过该功能的实现使学生理解类设计、特别是方法设计、变量命名等软件构造的理论知识。

二. 实验内容

1、培训计划信息显示，如下图所示。当用户点击培训成绩管理菜单时，系统按照时间顺序将当前所有培训计划信息显示到 JTable 中。

查询条件

培训计划年度：2015

培训专业：电气

培训计划类型：新员工入厂培训

完成情况：未培训

查询

共计：10条信息

序号	年度	培训计划名称	培训类型	培训专业	培训时间	完成情况	操作
1	2015	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	已培训	查看成绩
2	2015	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	已培训	查看成绩
3	2015	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	已培训	查看成绩
4	2015	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	已培训	查看成绩
5	2014	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	已培训	查看成绩
6	2014	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	已培训	录入成绩
7	2014	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	已培训	录入成绩
8	2014	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	已培训	录入成绩
9	2014	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	已培训	录入成绩
10	2014	新员工入厂培训	新员工入厂培训	电气 汽机 锅炉	1月10日-1月20日	已培训	录入成绩

2、培训成绩录入：用户点击上图中的【录入成绩】链接，显示出如下界面，将当前培训计划对应的每个每个学员信息展示在列表中。（不需要考虑考勤成绩）

a) 保存功能：将录入的成绩保存到数据表中。

b) 提交功能：除了实现保存数据功能外，还需要将培训计划表中状态字段进行修改

基本信息

年度	2015	名称	新员工入厂培训	时间	1月10日-1月20日
专业	电气	课时	3	授课人	吴浩

考勤/成绩信息

参加培训人数：30人 **总成绩 = 出勤成绩*20% + 考试成绩*80%**

序号	姓名	部门	出勤次数	出勤成绩	考试成绩	总成绩
1	张志刚	发电一部 电气1班	10			
2	李峰	发电一部 电气1班	10			
3	李峰	发电一部 电气1班	10			
4	李峰	发电一部 电气1班	10			
5	李峰	发电一部 电气1班	10			
6	王琦	发电一部 电气2班	10			
7	王琦	发电一部 电气2班	10			
8	王琦	发电一部 电气2班	10			
9	王琦	发电一部 电气2班	10			
10	王琦	发电一部 电气2班	10			

3、修改成绩：对于成绩未提交成绩的培训计划，可以允许修改成绩。

基本信息

年度	2015	名称	新员工入厂培训	时间	1月10日-1月20日
专业	电气	课时	3	授课人	吴浩

考勤/成绩信息

参加培训人数：30人 **总成绩 = 出勤成绩*20% + 考试成绩*80%**

序号	姓名	部门	出勤次数	出勤成绩	考试成绩	总成绩
1	张志刚	发电一部 电气1班	10	20	80	100
2	李峰	发电一部 电气1班	10	15	70	85
3	李峰	发电一部 电气1班	10	15	70	85
4	李峰	发电一部 电气1班	10	15	70	85
5	李峰	发电一部 电气1班	10			
6	王琦	发电一部 电气2班	10			
7	王琦	发电一部 电气2班	10			
8	王琦	发电一部 电气2班	10			
9	王琦	发电一部 电气2班	10			
10	王琦	发电一部 电气2班	10			

4、成绩查看：对于已经提交成绩的培训计划，可以允许查看成绩。

基本信息

年度	2015	名称	新员工入厂培训	时间	1月10日-1月20日
专业	电气	课时	3	授课人	吴浩

考勤/成绩信息

参加培训人数：30人

总成绩 = 出勤成绩*20% + 考试成绩*80%

导出

打印

序号	姓名	部门	出勤次数	出勤成绩	考试成绩	总成绩
1	张志刚	发电一部 电气1班	10	20	80	100
2	李峰	发电一部 电气1班	10	15	70	85
3	李峰	发电一部 电气1班	10	15	70	85
4	李峰	发电一部 电气1班	10	15	70	85
5	李峰	发电一部 电气1班	10	15	70	85
6	王琦	发电一部 电气2班	10	15	70	85
7	王琦	发电一部 电气2班	10	15	70	85
8	王琦	发电一部 电气2班	10	15	70	85
9	王琦	发电一部 电气2班	10	15	70	85
10	王琦	发电一部 电气2班	10	15	70	85

三. 实验过程及结果

- 1、按照实验要求完成相应功能
- 2、打印该功能的源代码，以及测试结果；

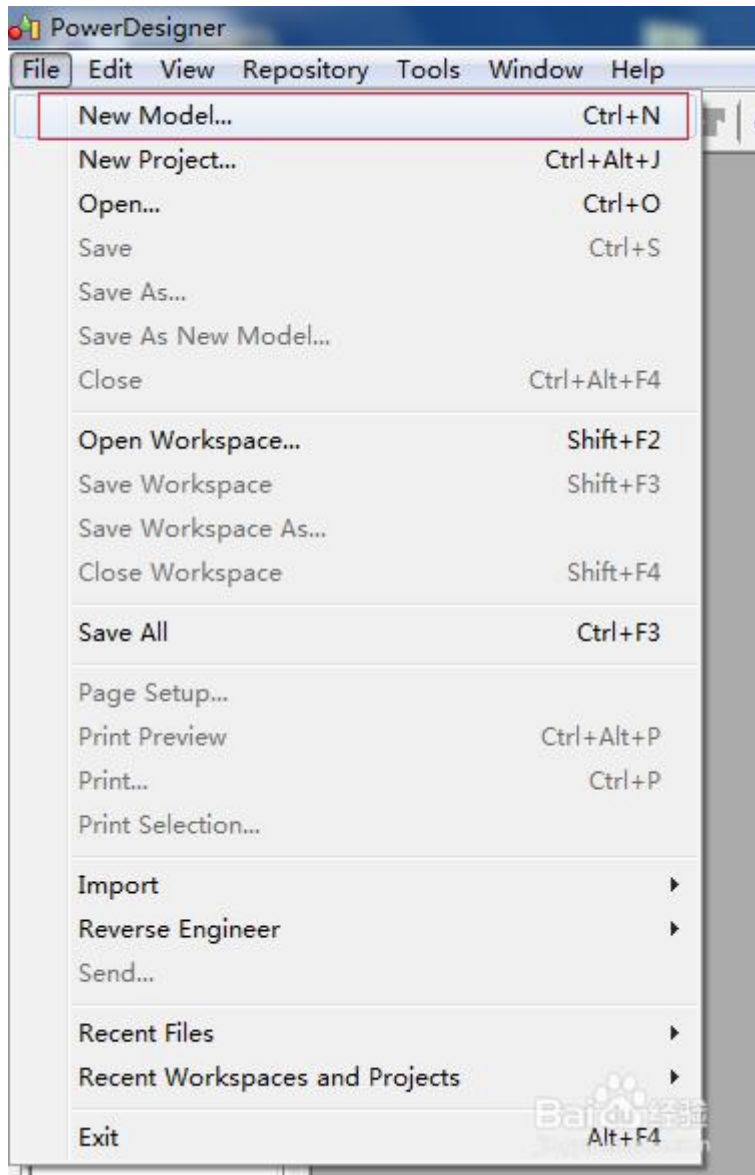
四. 实验遇到问题及体会（必须要写）

- 1、在本次实验中你应用到哪些软件构造的知识，并举例说明。
- 2、在实现过程遇到了什么问题，以及如何解决的？
- 3、讲讲本次实验的收获。

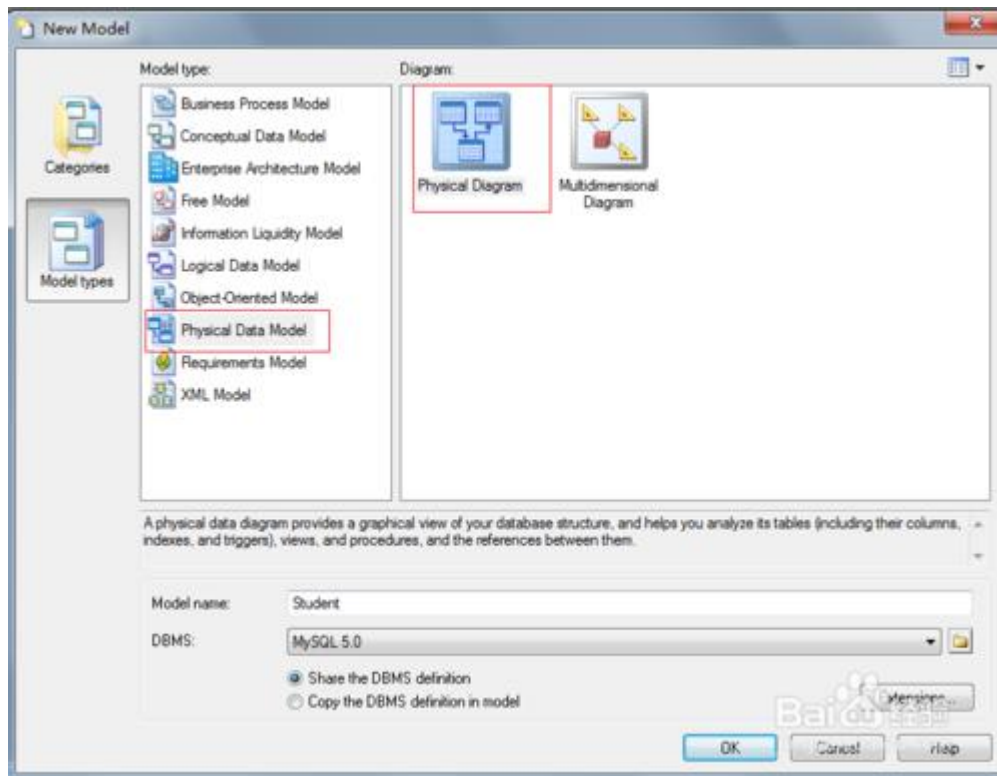
附件：

二、Power Designer 创建 PDM（物理模型）教程

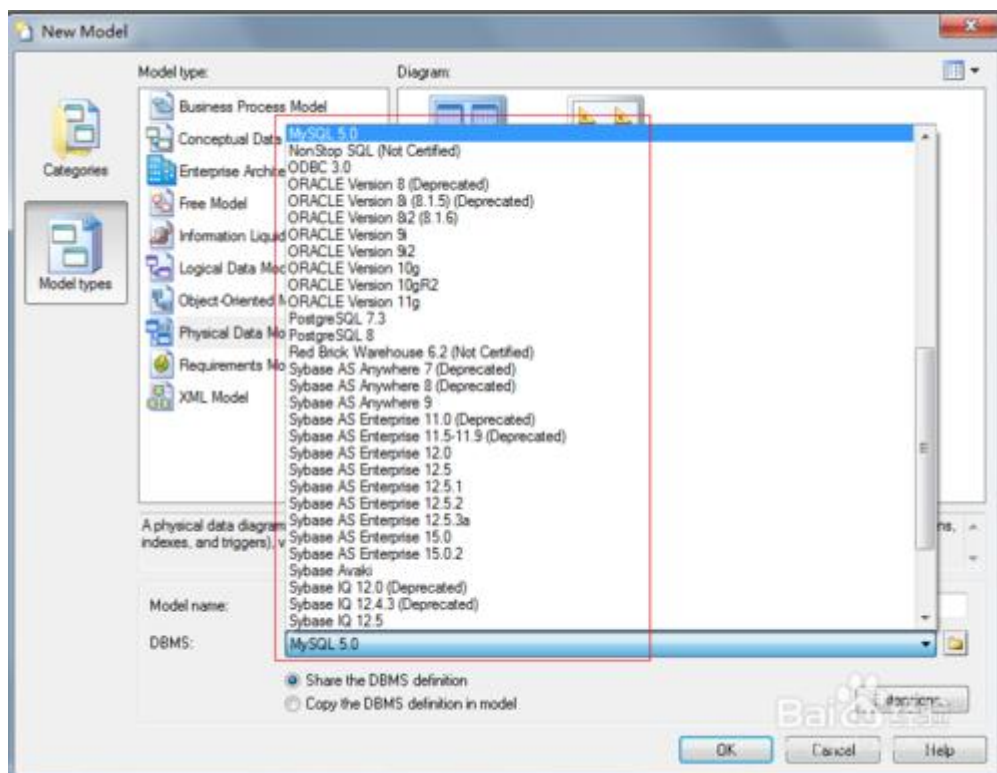
- 点击菜单栏新建



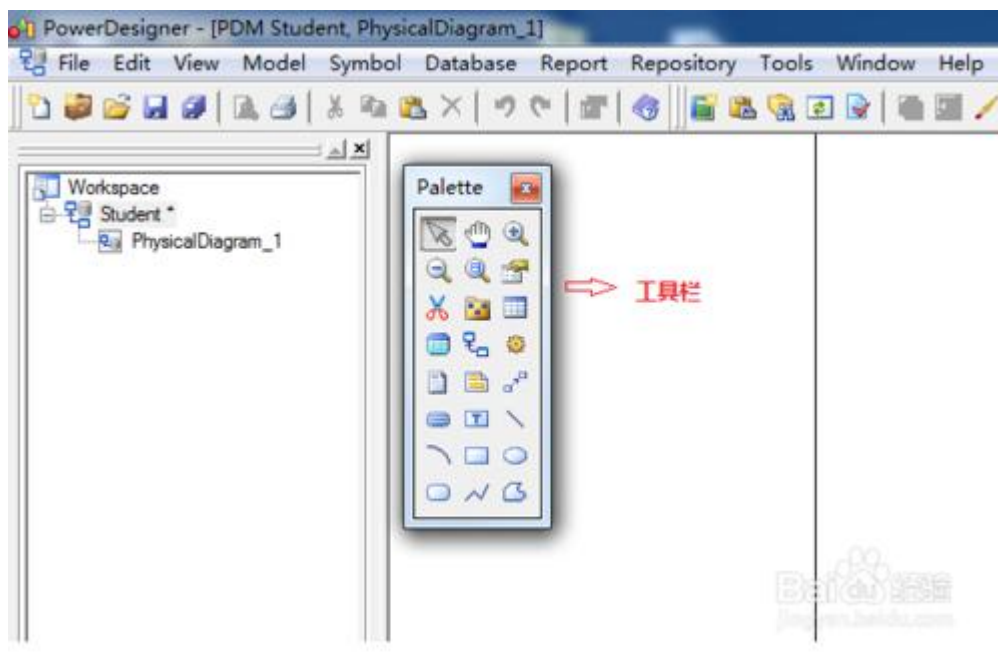
- 选择新建物理数据对象模型



- 选择使用数据库

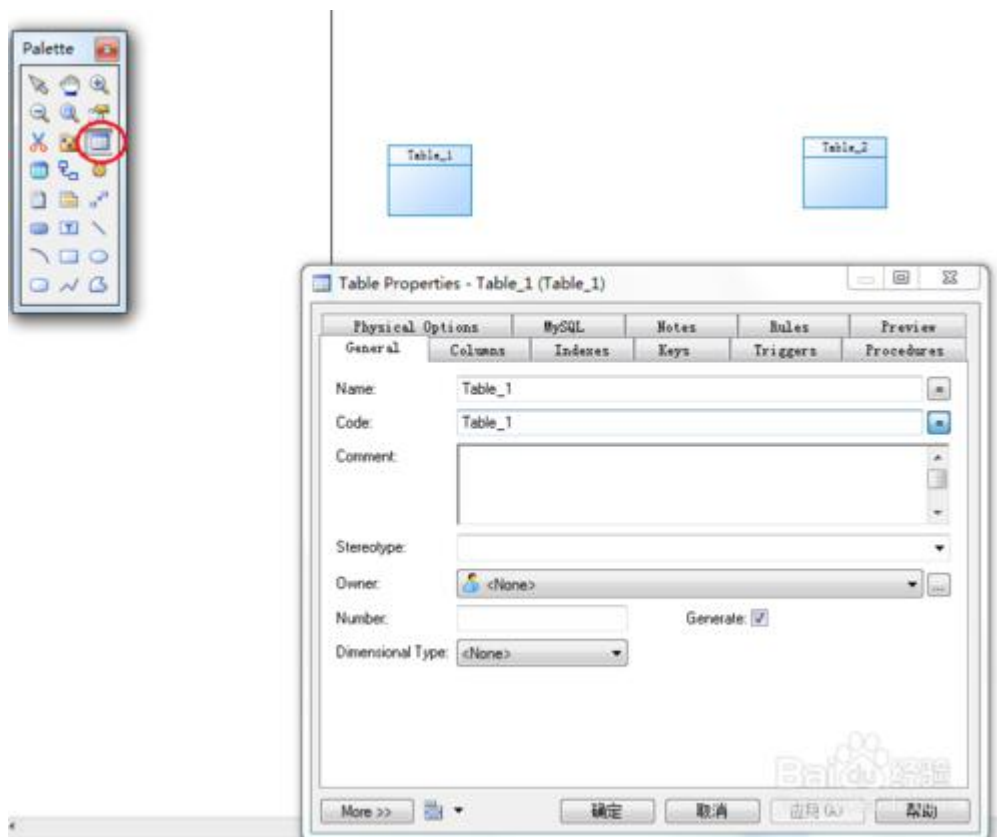


- 给新建的物理对象模型命名，点击确定，进入物理对象模型编辑界面
- 图中标记工具栏，读者可对每个工具做深入学习

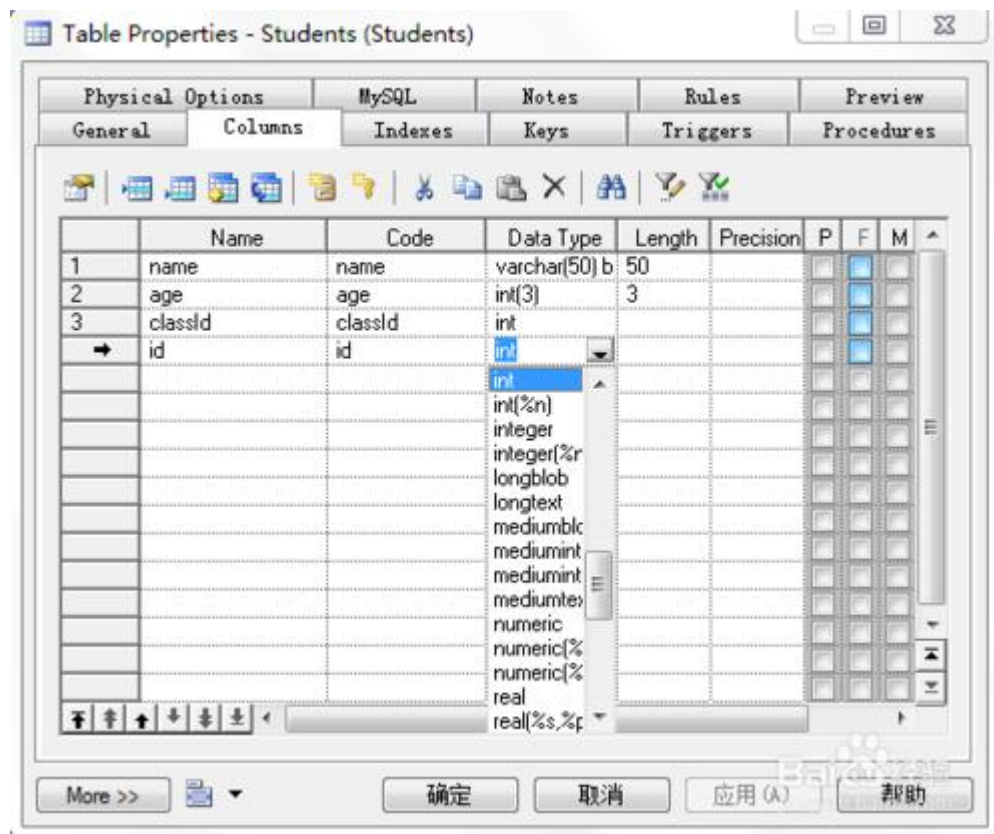
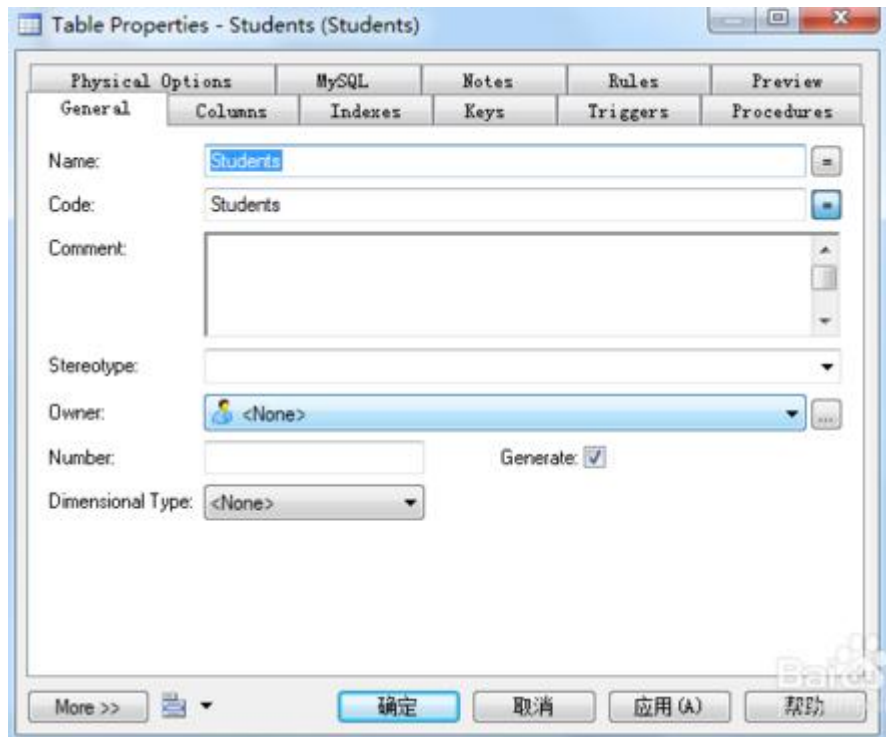


- 本文提供一个简单示例，新建两张表（Students，Class）并提供一个外键约束
具体步骤：

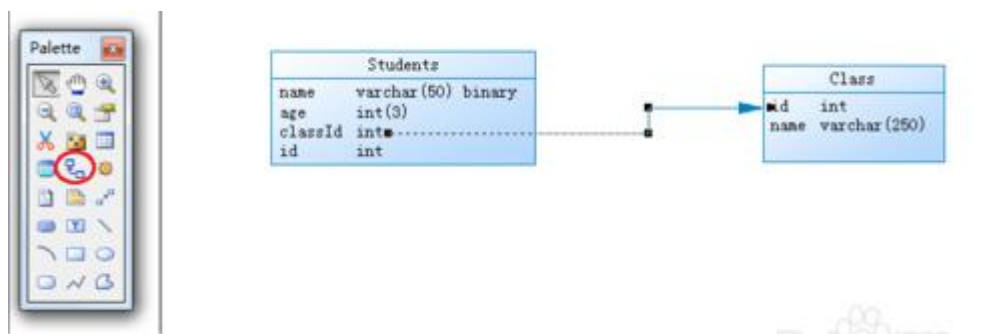
(1) 点击工具，新建两张表



(2) 双击新建好的表，进入 table 编辑界面，指定表名，字段等属性



(3)增加外键，Students 表的 classId 指向 class 表的 id 字段



三、系统数据库物理模型初步设计

1. 培训计划基本信息表: 计划 ID(主键) 名称 开始时间 结束时间 创建时间
培训计划类型 年度 完成情况;

2. 培训计划详细信息表: ID(主键) 计划 ID(外键) 专业 ID 培训目的 培训内容 课时 授课人;

3. 部门表: ID(主键) 部门名称 部门编号 备注 上级部门编号 联系方式

0	二电厂	000	备注 1	-1
1	部门 1	001	备注 1	0
2	部门 2	002	备注 1	0
3	部门 11	0011	备注 1	1

4. 员工表: ID(主键) 姓名 性别 身份证 出生日期 部门编号 岗位 职务 籍贯
照片 联系电话

a) 照片字段设计方式: (1) 类型设计为二进制, 直接将内容保存到字段中;

(2) 类型设计为 varchar, 字段中保存的是图片的路径;

5. 员工参加培训表: 部门 ID 员工 ID 培训计划 ID 成绩 成绩状态 (1: 已提交 0: 未提交)

6. 用户表: ID 用户名 密码 是否启用 (1: 启用 0: 未启用)

7. 角色表: ID 角色名称 角色描述 角色权限

8. 用户角色表: ID 用户 ID 角色 ID

9. 字典表: ID 字典名称 字典编号 描述 上级字典;

1	培训计划类型	001	0
2	新员工培训	00101	1

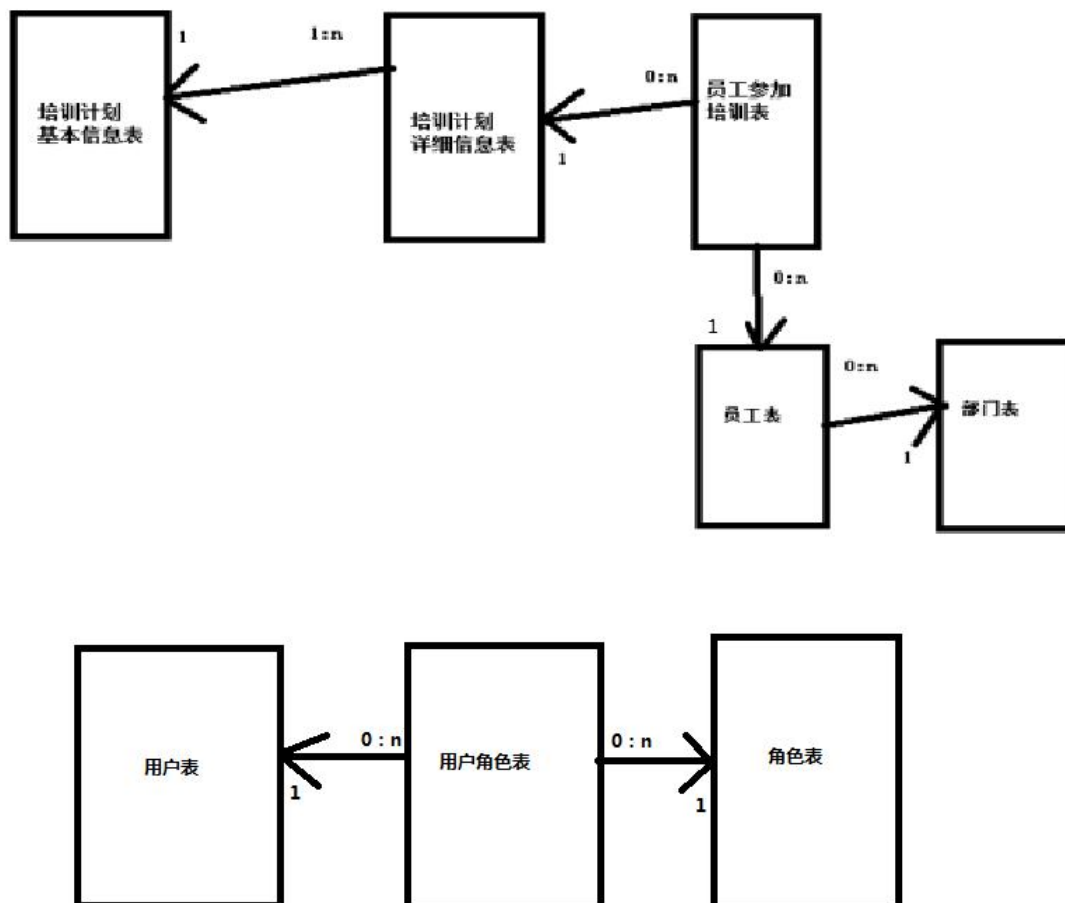
系统中的字典有:

(1) 培训计划类型 : 新员工培训、 中层领导培训

(2) 培训专业：电气、 热工

(3) 岗位：厂长、副厂长、车间主任、副主任、班长、普通员工

(4) 职称：高级工程师 工程师 助理工程师 技师



四、WindowBuilder 离线安装过程

WindowBuilder 是可视化 Java GUI 编程的 eclipse 插件。有了它的帮助，我们可以通过拖拽来编辑 Java 程序界面。安装过程如下：

1. 下载 WindowBuilder

<http://eclipse.org/windowbuilder/download.php>

在上面网站选择与自己 eclipse 版本相符的 WindowBuilder。如下图所示：

Update Sites

Eclipse Version	Release Version		Integration Version	
	Update Site	Zipped Update Site	Update Site	Zipped Update Site
4.3 (Kepler)			link	link (MD5 Hash)
4.2 (Juno)	link	link (MD5 Hash)	link	link (MD5 Hash)
3.8 (Juno)	link	link (MD5 Hash)	link	link (MD5 Hash)
3.7 (Indigo)	link	link (MD5 Hash)	link	link (MD5 Hash)
3.6 (Helios)	link	link (MD5 Hash)	link	link (MD5 Hash)
3.5 (Galileo)	link	link (MD5 Hash)	link	link (MD5 Hash)
3.4 (Ganymede)	link	link (MD5 Hash)		

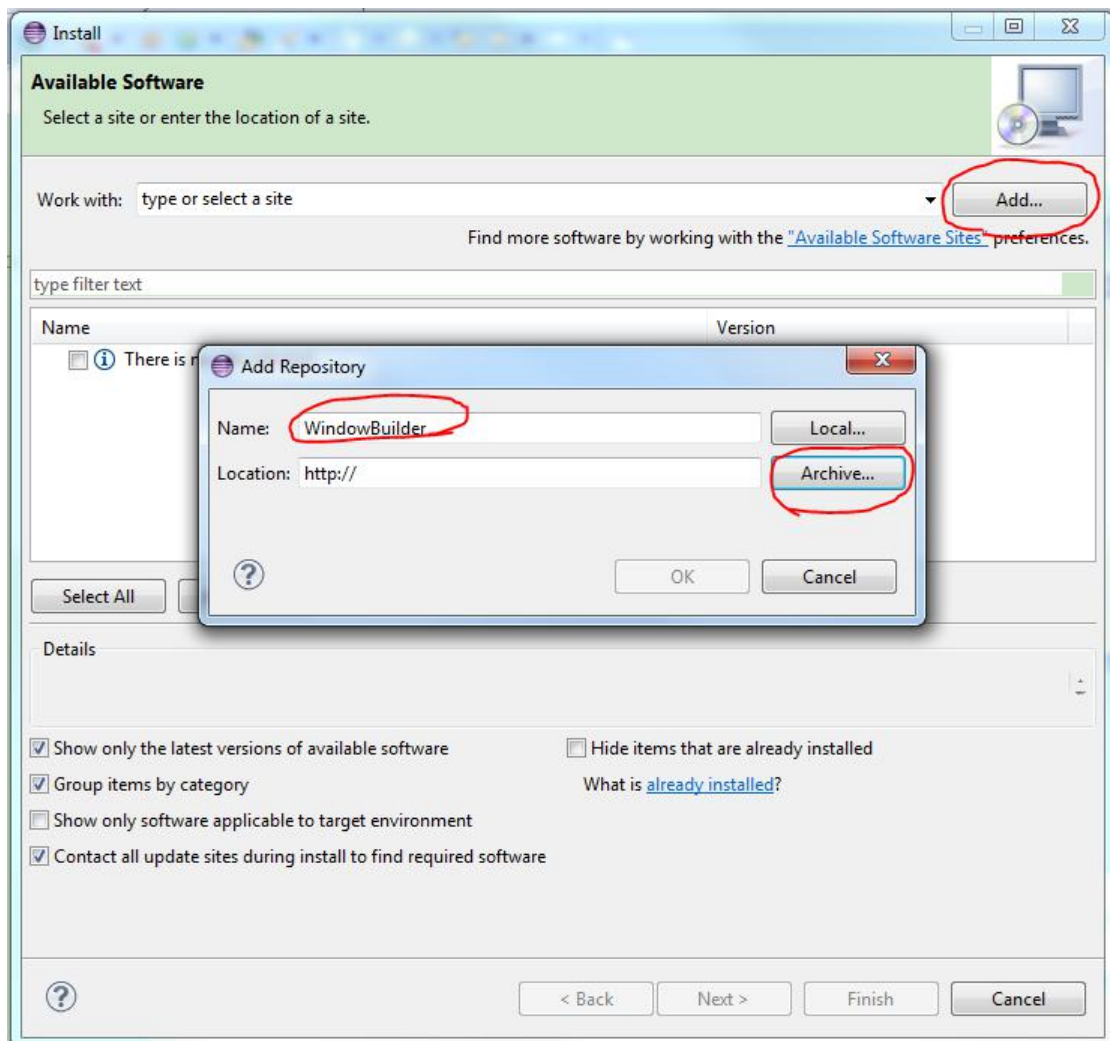
我安装的是 eclipse4.2 所以下载红笔所示的版本。下载后的.zip 文件如下图：



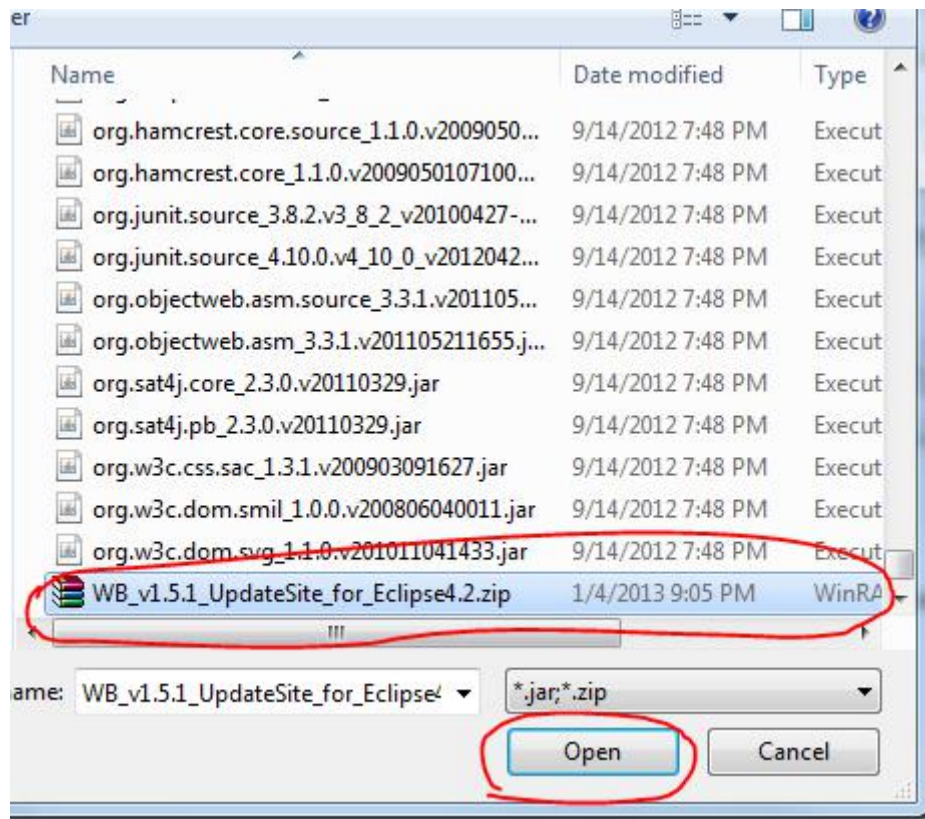
2.安装 WindowBuilder

1) 把上图所示的.zip 文件拷贝到 eclipse 安装目录下 plugins 文件夹里。注意：是把整个 zip 文件，不需要解压。

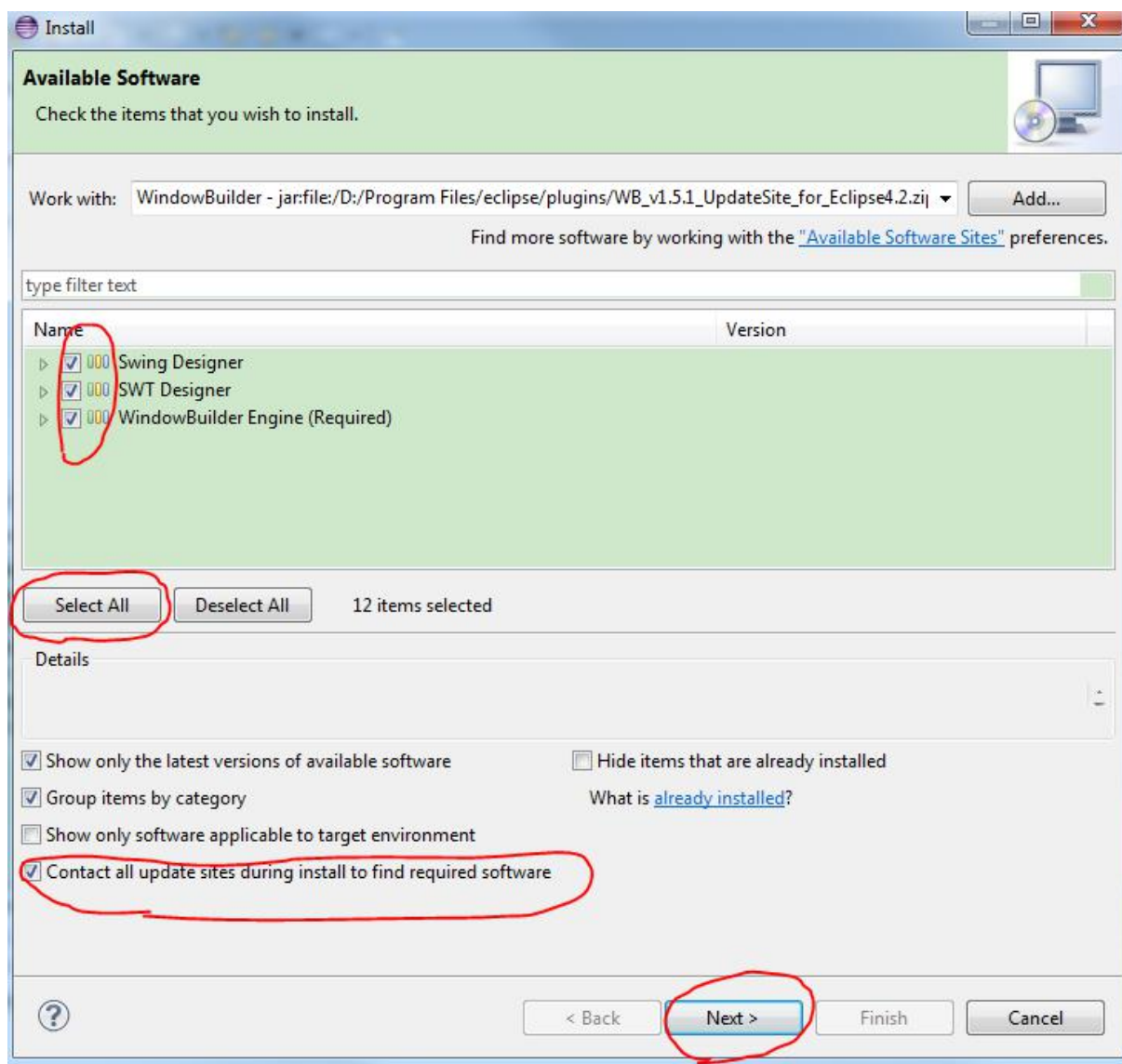
2) 打开 eclipse -- Help -- Install New Software... -- Add... -- Archive...



在弹出的对话框中选择之前放进 plugins 文件夹的 zip 文件，Open

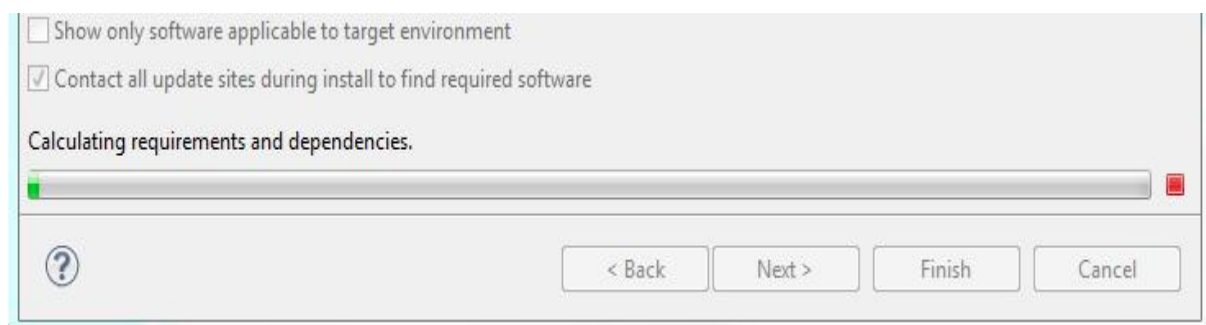


然后点 OK，会看到

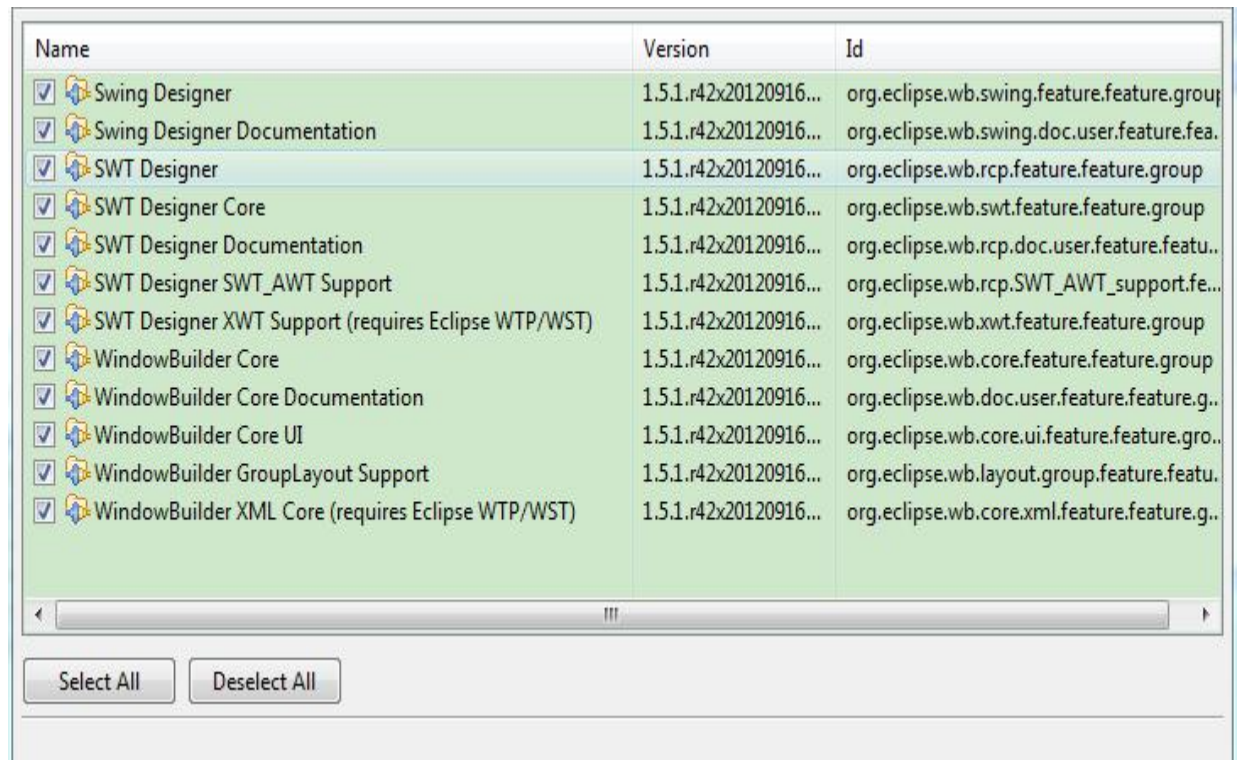


选择 Select All, 然后点击 Next.

这时会看见下面的画面, 需要一些耐心来等待 完成。

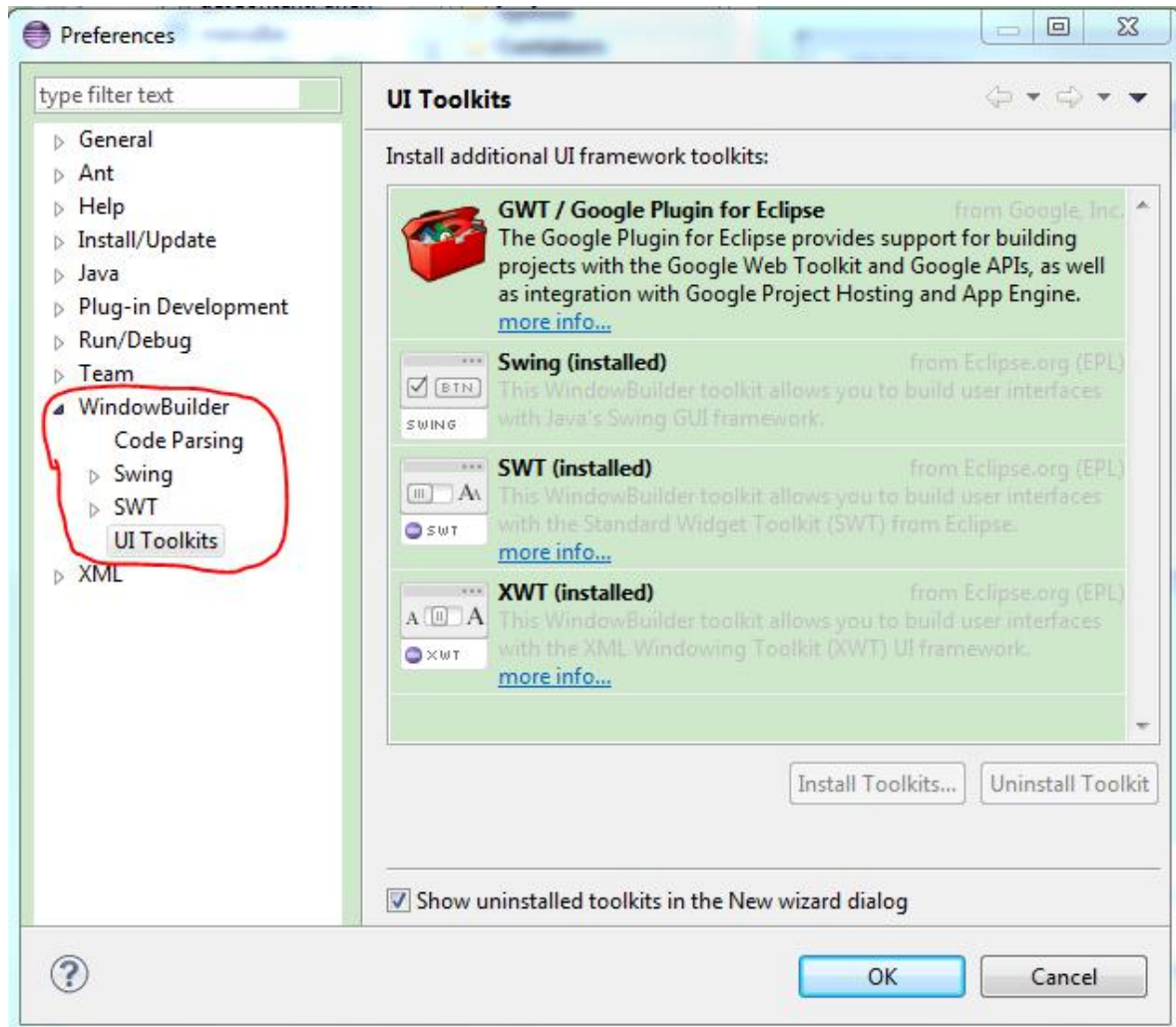


等下载完所需文件之后, 会看到这样的情况



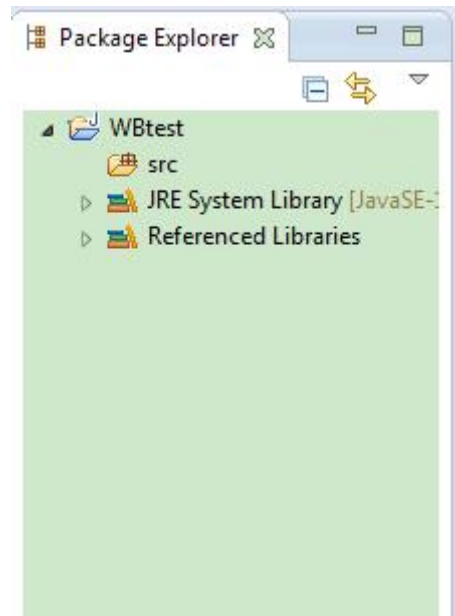
之后一路点下去就好了，会有一个安装过程，安装完后弹出一个对话框提示重新启动 eclipse。

重新启动 eclipse 后，在 Window -- Preferences 看到下面情况，说明安装成功。

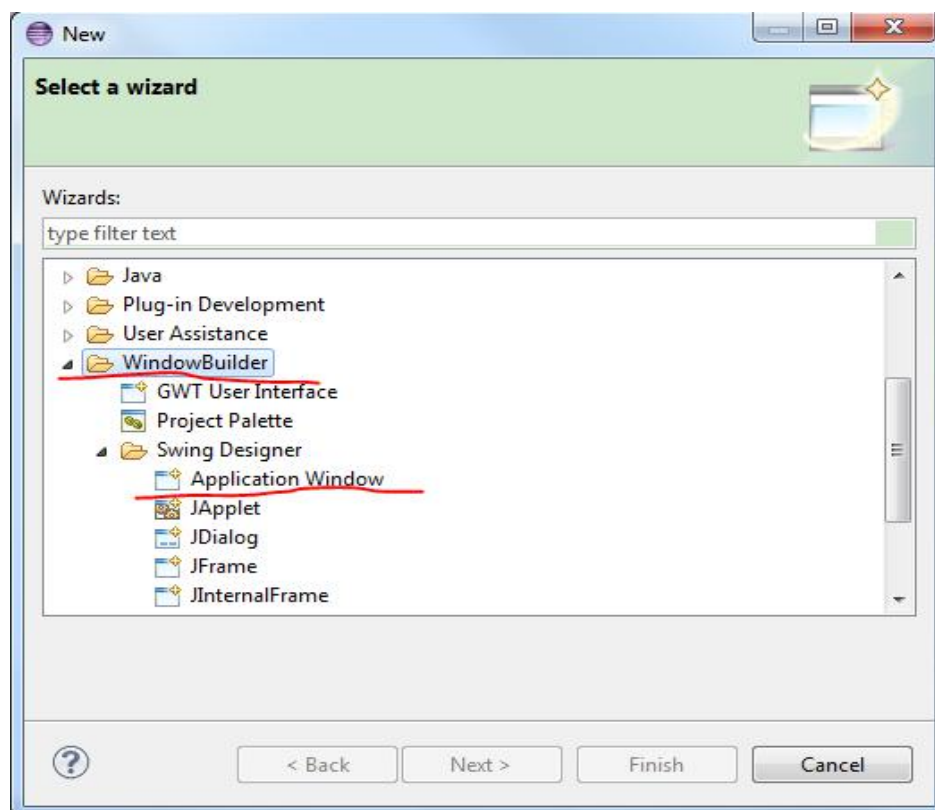


五、WindowBuilder 使用

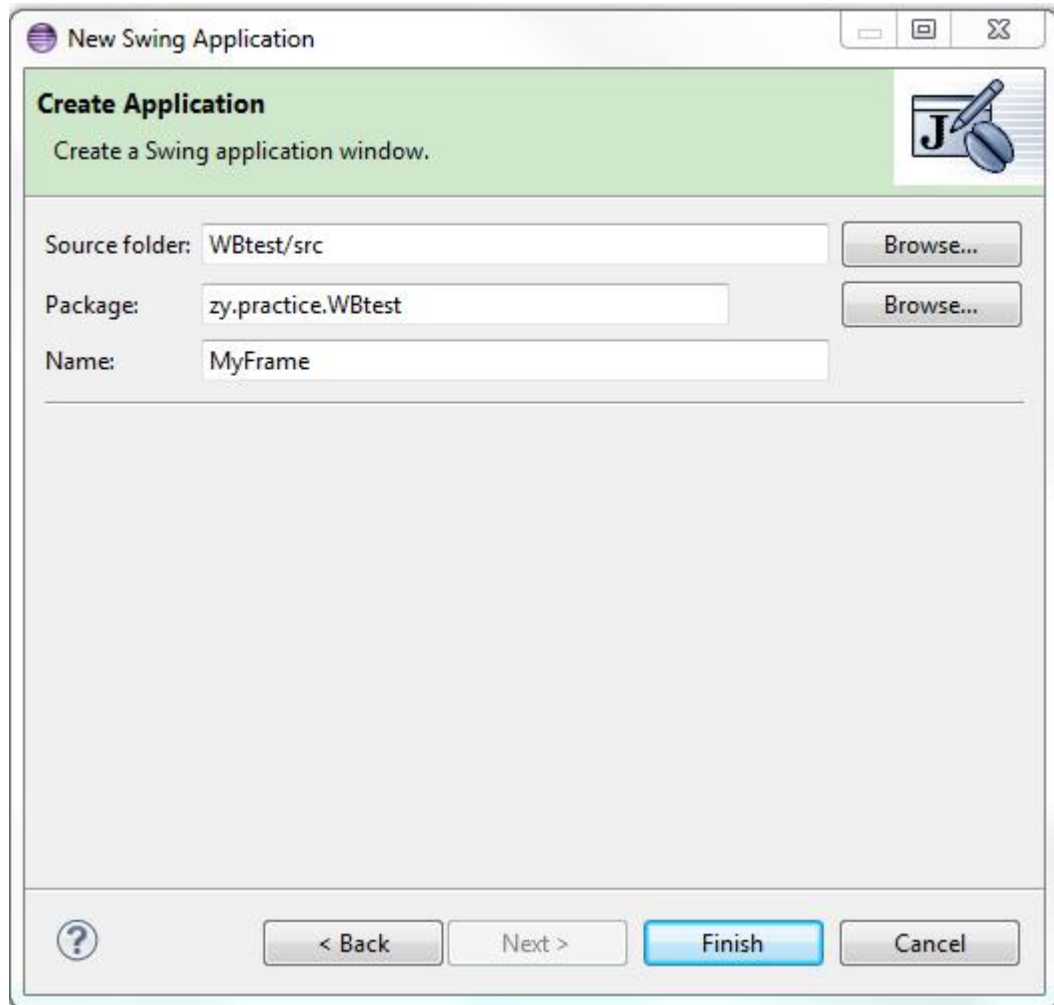
1. 新建一个 Java project. 取名为 WBtest, 如图所示:



2. 在 src 上右键 -- New -- Other... -- WindowBuilder -- Swing Designer -- Application Window



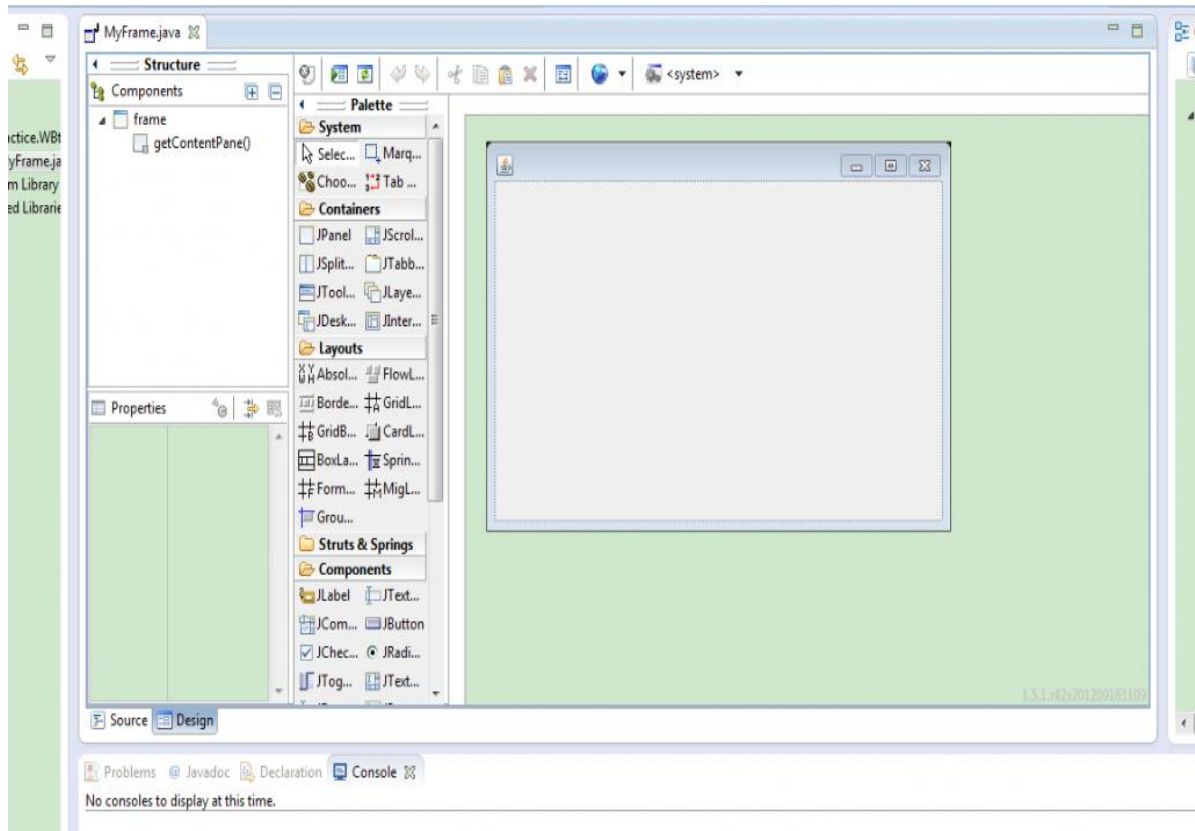
Next -- 输入 Package 和 Name -- Finish



会看到下面画面：



注意红线圈出来的位置，点击 Design



可视化编程界面就出现了，这以后就可以根据自己的需要选择设计界面了。

六、DAO 设计模式

1、DAO 简介：

DAO 设计模式是属于 J2EE 数据层的操作，使用 DAO 设计模式可以简化大量代码，增强程序的可移植性。

2、DAO 各部分详解：

DAO 设计模式包括以下 5 个主要部分：

（1）、数据库连接类；

数据库连接类的主要功能是连接数据库并获得连接对象，以及关闭数据库。通过数据库连接类可以大大的简化开发，在需要进行数据库连接时，只需常见该类的实例，并调用其中的方法就可以获得数据库连接对象和关闭数据库，不必再进行重复操作。例：

```
Package com.javaweb.dao1

Import java.sql.*;

Public class DataBaseConnection{

    Private final String DBDRIVER = “com.mysql.jdbc.Driver” ;

    Private final String DBURL = “jdbc:mysql://localhost:3306/javaweb” ;

    Private final String DBUSER = “root” ” ;

    Private final String DBPASSWORD = “585762” ;

    Private Connection conn = null;

    Public DataBaseConnection(){

        Try{

            Class.forName(DBDRIVER);
            This.conn =

DriverManager.getConnection(DBURL,DBUSER,DBPASSWORD);
        }catch(Exception e){
            System.out.println(“加载数据库驱动失败！ ” );
        }

    }

}
```

```

    Public Connection getConnection(){
        Return conn;
    }

    Public void close(){
        Try{
            Conn.close();
        }catch(){
            System.out.println(“数据库连接关闭失败！”);
        }
    }
}

```

(2)、VO 类:

VO 类是一个包含属性和表中字段完全对应的类,并在该类中提供 setter 和 getter 方法来设置并获取该类中的属性。例:

```

Package com.javaweb.dao1
Public classs User{
    Private int userid;
    Private String username;
    Private String password;
    Public int getUserid(){
        Return userid;
    }
    Public void setUserid(int userid){
        Return userid;
    }
    Public String getUsername(){
        Return username;
    }
    Public void setUsername(String username){
        Return username;
    }
}

```

```

    Public String getUserpassword(){
        Return userpassword;
    }
    Public void setUserid(String userpassword){
        Return userpassword;
    }
}

```

(3)、DAO 接口：

DAO 接口中定义了所有的用户操作，如添加记录、删除记录及查询记录等。不过因为是借口，所以仅仅是定义，需要子类实现。例：

```

Package com.javaweb.dao1;
Import java.util.*;
Public class UserDAO{
    public void insert(User user)throw Exception;
    Public void update(User user)throw Exception;
    Public void delete(int userid)throw Exception;
    Public User queryById(int userid)throw Exception;
    Public List queryAll()throw Exception;
}

```

(4)、DAO 实现类：

DAO 实现类实现了 DAO 接口，并实现了接口中定义的所有方法。

(5)、DAO 工厂类：

在没有 DAO 工厂类的情况下，必须通过创建 DAO 实现类的实例才能完成数据库操作。这时就必须知道具体的子类，对于后期的修改非常不方便。

使用 DAO 工厂类，可以比较方便地对代码进行管理，而且可以很好地解决后期修改的问题，通过该 DAO 工厂类的一个静态方法来获取 DAO 实现类实例。这时如果要替换 DAO 实现类，只需要修改该 Dao 工厂类中的方法代码，而不必邀修改所有的操作数据库代码。例：

```
Package com.javaweb.daol;  
  
Public class DAOFactory{  
  
    Public static UserDAO getUserDAOInstance() {  
  
        Return new UserDAOImpl();  
  
    }  
  
}
```

3、使用 DAO 完成数据库操作：

- (1)、添加记录：
- (2)、更新记录：
- (3)、删除记录：
- (4)、查询记录：

七、Apache DBUtils 框架

1. commons-dbutils 简介

commons-dbutils 是 Apache 组织提供的一个开源 JDBC 工具类库，它是对 JDBC 的简单封装，学习成本极低，并且使用 dbutils 能极大简化 jdbc 编码的工作量，同时也不会影响程序的性能。因此 dbutils 成为很多不喜欢 hibernate 的公司的首选。

commons-dbutils API 介绍：

- org.apache.commons.dbutils.QueryRunner
- org.apache.commons.dbutils.ResultSetHandler

工具类

- org.apache.commons.dbutils.DbUtils

2. QueryRunner 类使用讲解

该类简单化了 SQL 查询，它与 ResultSetHandler 组合在一起使用可以完成大部分的数据库操作，能够大大减少编码量。

QueryRunner 类提供了两个构造方法：

- 默认的构造方法
- 需要一个 javax.sql.DataSource 来作参数的构造方法。

2.1、QueryRunner 类的主要方法

public Object query(Connection conn, String sql, Object[] params, ResultSetHandler rsh) throws SQLException: 执行一个查询操作，在这个查询中，对象数组中的每个元素值被用来作为查询语句的置换参数。该方法会自行处理 PreparedStatement 和 ResultSet 的创建和关闭。

public Object query(String sql, Object[] params, ResultSetHandler rsh) throws SQLException: 几乎与第一种方法一样；唯一的不同在于它不将数据库连接提供给方法，并且它是从提供给构造方法的数据源 (DataSource) 或使用的 setDataSource 方法中重新获得 Connection。

public Object query(Connection conn, String sql, ResultSetHandler rsh) throws SQLException: 执行一个不需要置换参数的查询操作。


```
public int update(Connection conn, String sql, Object[] params) throws
```

SQLException:用来执行一个更新（插入、更新或删除）操作。

public int update(Connection conn, String sql) throws SQLException: 用来执行一个不需要置换参数的更新操作。

2.2、使用 QueryRunner 类实现 CRUD

```
package me.gacl.test;
import java.util.Date;
import java.util.List;
import java.io.File;
import java.io.FileReader;
import java.io.IOException;
import java.sql.SQLException;
import javax.sql.rowset.serial.SerialClob;
import me.gacl.domain.User;
import me.gacl.util.JdbcUtils;
import org.apache.commons.dbutils.QueryRunner;
import org.apache.commons.dbutils.handlers.BeanHandler;
import org.apache.commons.dbutils.handlers.BeanListHandler;
import org.junit.Test;
/**
 * @ClassName: DBUtilsCRUDTest
 * @Description:使用 dbutils 框架的 QueryRunner 类完成 CRUD, 以及批处
理
 */
public class QueryRunnerCRUDTest {
    /*
    *测试表
    create table users(
    id int primary key auto_increment,
    name varchar(40),
    password varchar(40),
    email varchar(60),
    birthday date
    );
    */
    @Test
    public void add() throws SQLException {
        //将数据源传递给 QueryRunner, QueryRunner 内部通过数据源获取数据库
        连接
        QueryRunner qr = new QueryRunner(JdbcUtils.getDataSource());
        String sql = "insert into users(name,password,email,birthday)
```

```

values(?,?,?,?)");
    Object params[] = {"孤傲苍狼", "123", "gacl@sina.com", new Date()};
    //Object params[] = {"白虎神皇", "123", "gacl@sina.com",
"1988-05-07"};
    qr.update(sql, params);
}
@Test
public void delete() throws SQLException {
    QueryRunner qr = new QueryRunner(JdbcUtils.getDataSource());
    String sql = "delete from users where id=?";
    qr.update(sql, 1);
}
@Test
public void update() throws SQLException {
    QueryRunner qr = new QueryRunner(JdbcUtils.getDataSource());
    String sql = "update users set name=? where id=?";
    Object params[] = { "ddd", 5};
    qr.update(sql, params);
}
@Test
public void find() throws SQLException {
    QueryRunner qr = new QueryRunner(JdbcUtils.getDataSource());
    String sql = "select * from users where id=?";
    Object params[] = {2};
    User user = (User) qr.query(sql, params, new
BeanHandler(User.class));
    System.out.println(user.getBirthday());
}
@Test
public void getAll() throws SQLException {
    QueryRunner qr = new QueryRunner(JdbcUtils.getDataSource());
    String sql = "select * from users";
    List list = (List) qr.query(sql, new
BeanListHandler(User.class));
    System.out.println(list.size());
}
/**
 * @Method: testBatch
 * @Description:批处理
 * @Author:孤傲苍狼
 *
 * @throws SQLException
 */
@Test
public void testBatch() throws SQLException {

```

```

    QueryRunner qr = new QueryRunner(JdbcUtils.getDataSource());
    String sql = "insert into users(name,password,email,birthday)
values(?,?,?,?)";
    Object params[][] = new Object[10][];
    for (int i = 0; i < 10; i++) {
        params[i] = new Object[] { "aa" + i, "123", "aa@sina.com",
        new Date() };
    }
    qr.batch(sql, params);
}
//用 dbutils 完成大数据（不建议用）
/*****
*****

create table testclob
(
id int primary key auto_increment,
resume text
);
*****
*****/

@Test
public void testclob() throws SQLException, IOException{
    QueryRunner runner = new QueryRunner(JdbcUtils.getDataSource());
    String sql = "insert into testclob(resume) values(?)"; //clob
    //这种方式获取的路径，其中的空格会被使用"%20"代替
    String path =
QueryRunnerCRUDTest.class.getClassLoader().getResource("data.txt")
.getPath();
    //将"%20"替换回空格
    path = path.replaceAll("%20", " ");
    FileReader in = new FileReader(path);
    char[] buffer = new char[(int) new File(path).length()];
    in.read(buffer);
    SerialClob clob = new SerialClob(buffer);
    Object params[] = {clob};
    runner.update(sql, params);
}
}
}

```