

## E 3

### Group K

#### 主题: Adapter View

#### 内容:

1. 修改 E 2 的 SCOS 工程 MainScreen 布局, 使用 GridView 替换已有的按钮导航: 点菜, 查看订单, 登录/注册, 系统帮助; 形成 launch board 导航样式。  
要求:
  - 1) GridView 中导航项为: 点菜, 查看订单, 登录/注册, 系统帮助
  - 2) 每个导航项都可点击
  - 3) 每个导航项标注有图标和文字说明, 图标自行设计, 采用 png 图片格式
  - 4) 当用户点击“登录/注册”选项时, 保持 E 2 原有功能实现
2. 在 SCOS 工程下新建 package 为: es.source.code.model
3. 在包 es.source.code.model 中新建一个 model 类 User.java, User 包含三个域: userName, password, oldUser, 数据类型分别为: String, String, Boolean, 每个域具有 Setter 和 Getter 方法
4. 修改 E 2 中 LoginOrRegister 代码, 完成如下功能:
  - 1) 在 LoginOrRegister 界面中添加新按钮“注册”
  - 2) 当用户在 LoginOrRegister 界面点击登录按钮时, 按照 E 2 验证规则验证用户输入是否正确。如果正确, 则新建 User 对象 loginUser, 将用户输入框输入的用户名和密码值通过 Setter 方法赋值到 loginUser 中, 并对其 oldUser 域赋值为 True, 屏幕跳转至 MainScreen, 并向 MainScreen 类传递数据 String 值为“LoginSuccess”, 同时将对象 loginUser 传递到 MainScreen 中。如果验证错误, 则按 E 2 中要求进行错误提示
  - 3) 当用户在 LoginOrRegister 界面点击注册按钮时, 按照 E 2 验证规则验证用户输入是否正确。如果正确, 则新建 User 对象 loginUser, 将用户输入框输入的用户名和密码值通过 Setter 方法赋值到 loginUser 中, 并对其 oldUser 域赋值为 False, 屏幕跳转至 MainScreen, 并向 MainScreen 类传递数据 String 值为“RegisterSuccess”, 同时将对象 loginUser 传递到 MainScreen 中。如果验证错误, 则按 E 2 中要求进行错误提示
  - 4) 当用户点击返回按钮时, 保持 E 2 中的功能不变
5. 修改 MainScreen.java 代码:
  - 1) 在 MainScreen 中信建一个 User 对象变量 user
  - 2) 判断由 LoginOrRegister 传回的 String 数据值。如果返回数据为“LoginSuccess”, 则保持 E 2 功能不变, 并将 user 赋值为 LoginOrRegister 传入的 loginUser
  - 3) 如果由 LoginOrRegister 传回的 String 数据值为“RegisterSuccess”, 则使用 Toast 提示用户“欢迎您成为 SCOS 新用户”。同时检查“点菜”和“查看订单”导

航项是否为隐藏状态，如果为隐藏，则设为显示，并将 user 赋值为 LoginOrRegister 传入的 loginUser

4) 当传入 mainScreen 的 String 值为其他时，将 user 赋值为 NULL

6. 在 SCOS 的 es.source.code.activity 包下新建 Activity 类名为: FoodView，为该 FoodView 定制布局 food\_view.xml，要求如下：

1) 在布局中使用 ViewPager & TabLayout 显示“冷菜”、“热菜”、“海鲜”、“酒水”四个导航目录

2) 支持屏幕左/右滑动导航

3) 导航至相应选项时，使用 ListView 或 RecyclerView 加载当前类别的食物列表，并显示到当前屏幕。要求 ListView 或 RecyclerView 每个选项含有：菜名，价格，点菜按钮。当点击点菜按钮时，使用 Toast 提示用户点菜成功，并将按钮显示为“退点”。当点击选项其他区域时，进入到菜品详细页面

4) 使用 ActionBar 为 FoodView 添加选项菜单，菜单项为“已点菜品”、“查看订单”、“呼叫服务”

7. 在 es.source.code.activity 包下新建 Activity 类名为: FoodOrderView，为该 FoodOrderView 定制布局 food\_order\_view.xml，要求如下：

1) 在该页面上使用 ViewPager & TabLayout 显示两个选项卡，分别为“未下单菜”和“已下单菜”

2) 支持屏幕左/右滑动导航，和导航项点击导航

2) 当用户导航至“已下单菜”，在 FoodOrderView 页面上方使用 ListView 或 RecyclerView 显示订单中的菜品项，每项中显示：菜名，价格，数量，备注；在页面下方显示菜品总数、订单总价及“结账”按钮

3) 当用户导航至“未下单菜”，在 FoodOrderView 页面上方使用 ListView 或 RecyclerView 显示已点但未生成订单菜品，每项显示：菜名，价格，数量，备注，“退点”按钮；在页面下方显示菜品总数、订单总价及“提交订单”按钮

8. 在 es.source.code.activity 包下新建 Activity 类名为: FoodDetailed，为该 FoodDetailed 定制布局 food\_detailed.xml，要求如下：

1) 在该布局中显示菜品信息：菜品图片，菜名，菜价，菜品备注输入框

2) 在该布局中添加一个按钮，按钮标签根据当前菜品是否已在已点菜品中，显示不同标签：如果是已点菜，则按钮显示“退点”，否则显示“点菜”

3) FoodDetailed 支持左/右滑动切换菜品。当用户在当前屏幕左/右滑动时，当前屏幕显示不同的菜品详细信息

9. 修改 FoodOrderView、FoodView 代码，在其中添加 User 变量 user

10) 修改 mainScreen 代码，完成如下功能：

1) 当用户点击“点菜”时，页面跳转至 FoodView，并使用 Intent 将 user 作为数据传给 FoodView 的 user 变量

2) 当用户点击“查看订单”时，页面跳转至 FoodOrderView，并使用 Intent 将 user 作为数据传给 FoodOrderView 的 user 变量

11. 修改 FoodView 代码，当用户点击食物列表项时，页面跳转至 FoodDetailed；当用户点击 ActionBar “已点菜品”时，页面跳转至 FoodOrderView，并默认显示为“未下单菜”选项页面，传入 user 变量；当用户点击 ActionBar “查看订单”时，页面跳转至 FoodOrderView，并默认显示为“已下单菜”选项页面，传入 user 变量
12. 修改 FoodOrderView 代码，当用户点击“结账”按钮时，当 user 不为 null 时，判断 user 中的 oldUser 值；如果为 True，则使用 Toast 提示“您好，老顾客，本次你可享受 7 折优惠”；否则，不作提示
13. 请根据以上任务，自主添加所需数据结构和功能代码，适当扩展功能实现，完成 SCOS 的基本原型开发，并调试正确运行
14. 修改 SCOS 工程的 AndroidManifest.xml 中 versionCode 和 versionName 为 2.0，打包发布 SCOS2.0.apk，并用真机测试运行

规则:

1. 在提交日期之前将成果提交至教学辅助系统
2. 提交文件格式为.doc 或.pdf
3. 文件命名格式为：学号+作业号。如 SA××××001+E 3.doc/pdf