

基本思路

1. 背景知识;
2. PyQt基本使用方法
 1. PyQt基本窗口对象;
 2. 布局
 3. 事件、信号与槽;
3. 常用开发方法;

1. 背景知识(5min)

可视化编程的含义

可视化编程，以“所见即所得”的编程思想为原则，力图实现编程工作的可视化，即化抽象为具体，将字符显示转化为图形信息，包括数据可视化和图形界面编程等。

图形用户界面（Graphical User Interface，简称 GUI，又称图形用户接口）是指采用图形方式显示的计算机操作用户界面。也即是我们平时看到的应用界面，最常见的GUI即是我们的桌面（操作系统界面）。

而常见的非图形界面编程，程序在命令窗口、Shell终端或其他终端上直接运行，用户与程序的交互只能通过字符输入输出的形式。

软件开发的基本流程

1. 业务分析

根据软件的使用情景，分析软件需要实现的功能，如开发一个考研背单词软件，那么我们需要实现的基本功能包括存取单词、标记单词以及翻译功能（你自己随便写）。

2. 程序设计

程序设计是指在宏观层面对代码进行一个大体设计，通常采用面向对象的思维分析设计对象及对象间的关系，使用伪代码或自然语言架构软件，具体将在下文展开。如老师上课点名，即包括老师，学生三个个体，其中老师对象具有点名功能（行为）。

3. 代码实现

代码实现即将伪代码或自然语言转化为相应的代码，通常会采用模块化开发的方式来组织代码文件，实现代码文件间的信息传递。

4. 调试维护

编写完代码后进行测试与调试，以得到符合要求的软件，同时软件的后期可维护性也是程序设计的重要指标。

Python的软件开发框架

Python常用的软件开发框架，或说是模块，这里展示的是三个主流开发模块，其中PyQt和PySide都是基于Qt开发的，PyQt更成熟一些，通常使用PyQt。

1. Tkinter
2. PyQt/PySide
3. wxpython

一般情况下，简单的软件使用tkinter即可完成，较复杂的软件通常采用PyQt或wxpython。（不展开介绍了，估计没时间）

2. 面向对象编程(15min)

基本窗口对象

首先，从宏观上了解认识各个窗口控件（带图介绍），然后展开介绍三个基本控件(QMainWindow, QLabel, QPushButton)的函数用法、参数含义（代码介绍），介绍如何查资料自学其他函数的用法，可以推荐一些相关的学习资料。

组成窗体的基本对象包括窗口控件、布局、信号与槽等等。

布局

布局表示的是上述控件之间的位置关系，包括绝对布局、相对布局和表格布局，表格布局分为水平布局、垂直布局与表格布局。

事件、信号与槽

事件，信号与槽是完成人机交互的关键，事件是用户的行为。

开发工具

PyQt designer提一嘴，有这个工具即可

3. 模块化编程(5min)

关于面向对象编程和模块化编程，你根据内容，合并到一个PPT，简述其思想即可，不用展开介绍，我没做完的。完成界面逻辑后，如何运用类开发来实现主程序、业务程序与界面程序的信息传递。（该内容对基础有一定的要求，简单描述即可）

主程序如何调起各模块，实现模块间的信息传递与软件的功能实现，主要分为单类继承和多类继承两种方法。

（提一下就算了，或者可以直接把这一面PPT删了，这里本来就比较复杂，到时候他们问起来反而尴尬）

补充

1. 结合Pandas、Database、Matplotlib等的应用；
2. Python并不是专业的软件开发语言；

简单综述

自己简单概括一下

我设计的整体思路是，第二板块是主要内容，第一板块是基本思想，第三板块是附加提高工具，若听众多数对编程不熟悉的人而言，应侧重于第二板块，对于他们更实用，我给的建议也是重心第二板块的介绍，即基本对象介绍、基本布局和事件信号和槽的介绍，考虑到时间关系，其中又以基本对象介绍为主，只介绍最基本的两种QMainWindow和QLabel即可，但是一定要讲清楚用法，整体根据听众水平和你的情况安排时间，第一部分和第三部分简单带过即可，不要深入展开，部分专业术语改成你的口语，用你的理解去陈述，本身就是一些比较深的东西。