

CI 103 Introduction to Computing and Information Security

*Project Proposal Presentation for*

# Economy Fueler

- 01** Overview of the Project
- 02** Brief Introduction of the Work
- 03** Technology Sharing 1: Framework Selection
- 04** Technology Sharing 2: Django MTV
- 05** Technology Sharing 3: Database Design

# CONTENT



01

# Overview of the Project

The bottom half of the slide features a series of thin, light blue wavy lines that flow horizontally across the frame, creating a sense of movement and depth.

# Overview of the Project

01



## ECONOMY FUELER

### **Data**

Acquire the financial data from the enterprises.

### **Analysis**

Basic financial analysis based on the specific equations to evaluate their financial status.

### **Strategy**

Exhibit the analysis results and provide corresponding financial advices.



**02**

**Brief  
Introduction  
of Our Work**



**ECONOMY FUELER**

# Lixiao Yang

Role: Project Manager

02

- Using MySQL 8.0 to create the relational database
- Refining models through financial material
- Assist in the development of algorithms





# Jiahe Xie

Role: Technical Director

02

- Using Django 3.1.7(in PyCharm 2020 3.2 x64), JQuery 3.6.0 and MySQL 8.0 to develop the back end and interact with front end.
- Data visualization using Echart.
- Data transforming using Python.



# Aoyu Liu

Role: Front End Designer

02

- Using Django 3.1.7 (together with PyCharm 3.5) and bootstrap 3.4.1 to develop the website and enable it to interact with the users and back end.





**ECONOMY FUELER**

# Xinyue Zeng

Role: Database Designer

02

- Using joint table & E-R chart to design the relational database.
- Using MySQL 8.0 to create the relational database.



**ECONOMY FUELER**

# Huiru Chang

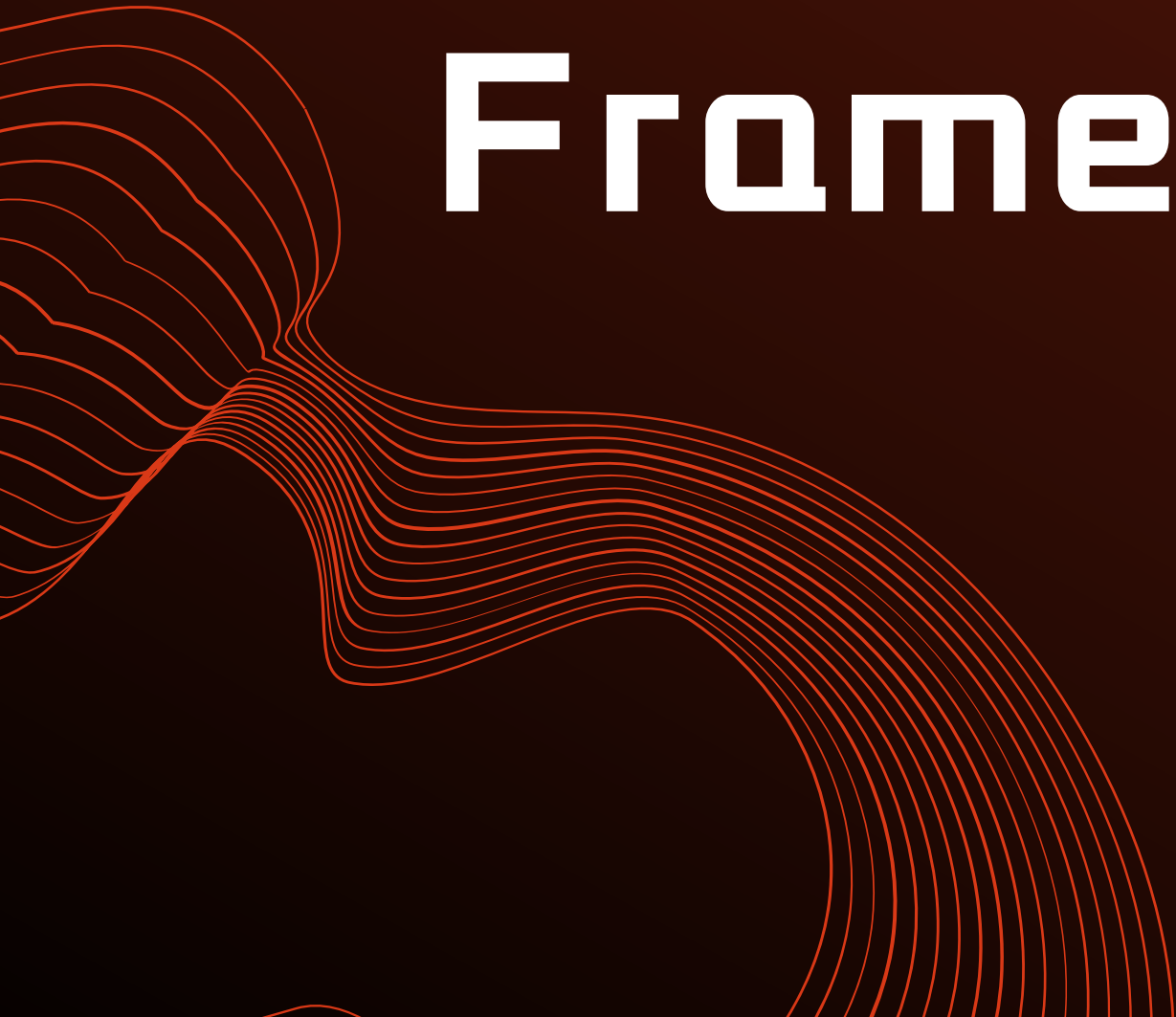
Role: UI Designer & Finance Modeler

02

- Using Dreamweaver to design and develop user interface.
- Assist in the financial modeling process.
- Assist in the data processing.

03

# Technology Sharing 1: Framework Selection



# Why we choose to use Django?

03

1. What we are learning is Python and its IDE PyCharm.
2. Why Django, not Flask or others?
3. Where we applied?



# Why we choose to use Bootstrap?

03

1. What is Bootstrap?
2. What benefits it brings?
3. Where we applied?



**Bootstrap**



04

# Technology Sharing 2: Django Features



# MTV in Django

04

## Model

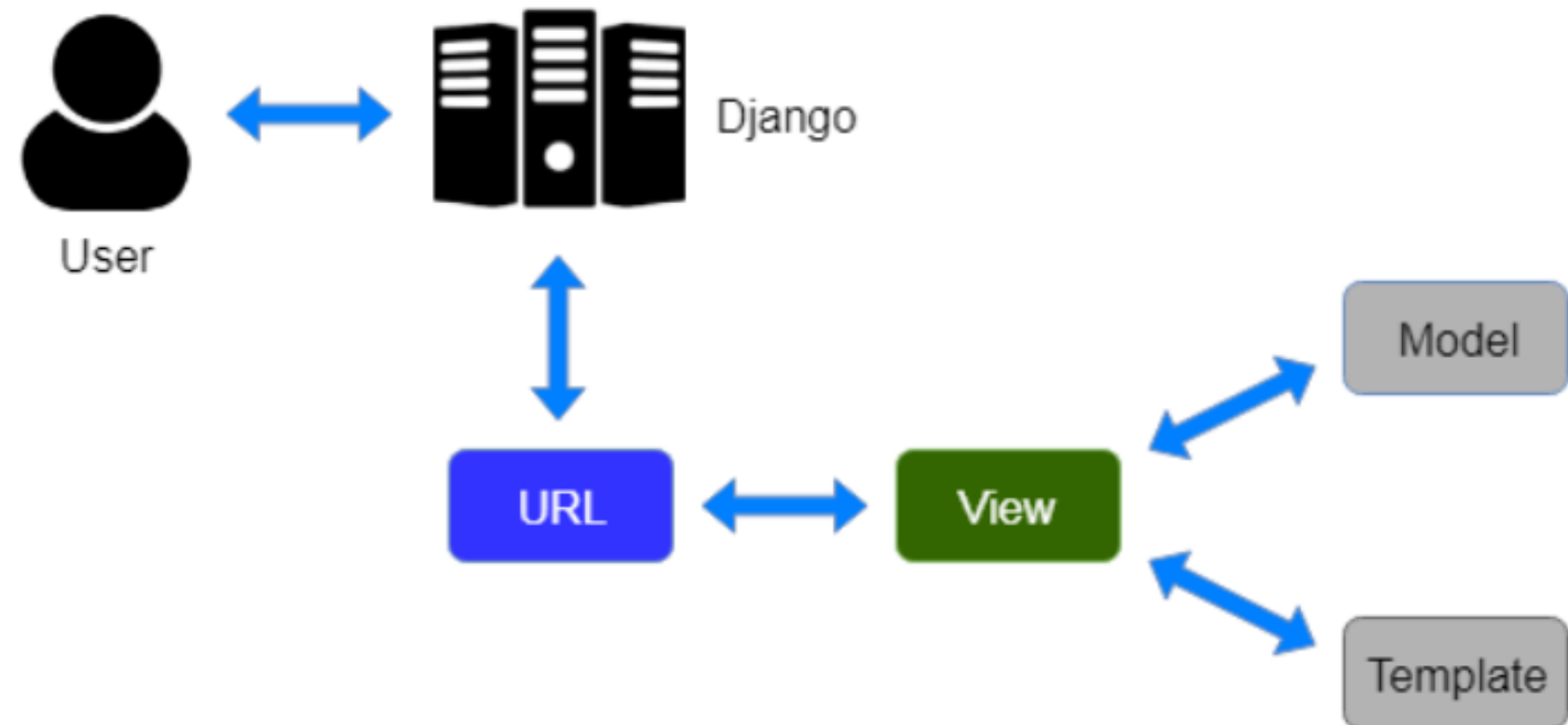
Interact with the database.

## Template

Return HTML.

## View

Accept the request and process the data(function).



# ORM in Django

04

Object Relational Mapping:

Object relational mapping (ORM, or O / RM, or O / R mapping) is a kind of programming technology, which is used to realize the data conversion between different types of systems in object-oriented programming language. In effect, it creates a "virtual object database" that can be used in programming language. Nowadays, there are many free and paid ORM products, and some programmers prefer to create their own ORM tools. 【×】

# ORM in Django

04

We can say it just does this thing.

```
conn = pymysql.connect(host='127.0.0.1', port=3306, user='root', passwd='a568178499',
db='babys', charset='utf8')
cursor = conn.cursor(cursor=pymysql.cursors.DictCursor)
cursor.execute('select id,title from class') -----> obj = models.class.objects.all()
result = cursor.fetchall()
cursor.close()
conn.close()
```

Transform the python statement into SQL statement and connect the database.

# Template Language

04

- Get the data from the back end.
- Inherit layout.

```
ecofuler > templates > background_index.html
1 {% extends 'layout.html' %} Inherit Parent Layout
2 {% block xx %}
3
4
5 <div style="..."> Extend Content
6   <table class="table" style="...">
7     <tr style="...">
8       <td colspan="4" style="..."><b style="...">访问量指标</b></td>
9       <td><button type="button" class="btn btn-default" style="...">2021-5-14</button></td>
10     </tr>
11     <tr>
12       <td style="...">PV<br><b style="...">100</b></td>
13       <td style="...">UV<br><b style="...">200</b></td>
14       <td style="...">IP<br><b style="...">300</b></td>
15       <td style="...">访客<br><b style="...">400</b></td>
16       <td style="...">内容总浏览量<br><b style="...">500</b></td>
17     </tr>
18   </table>
19 </div>
20 <div style="...">
21   <table class="table" style="...">
22     <tr>
23       <td colspan="4">
24         <a><i class="fa fa-envelope-o" aria-hidden="true"></i>邮件</a>
25       </td>
26     </tr>
27   </div>
28 </div>
29 <div class="pg-body">
30   <div class="menus">
31     <a href="/background_index/">首页</a>
32     <a href="/background_index/user_manage/">用户管理</a>
33     <a href="/background_index/database_manage/">数据库管理</a>
34   </div>
35   <div class="content">
36     <ol class="breadcrumb">
37       <li><a href="/background_index/">首页</a></li>
38       {% block li %}
39
40     {% endblock %}
41   </ol>
42   {% block xx %} Extend Content
43
44   {% endblock %}
45
```

```
ecofuler > templates > layout.html
25 <a><i class="fa fa-envelope-o" aria-hidden="true"></i>邮件</a>
26 </div>
27 </div>
28 </div>
29 <div class="pg-body">
30   <div class="menus">
31     <a href="/background_index/">首页</a>
32     <a href="/background_index/user_manage/">用户管理</a>
33     <a href="/background_index/database_manage/">数据库管理</a>
34   </div>
35   <div class="content">
36     <ol class="breadcrumb">
37       <li><a href="/background_index/">首页</a></li>
38       {% block li %}
39
40     {% endblock %}
41   </ol>
42   {% block xx %} Extend Content
43
44   {% endblock %}
45
```

# Reload

04

Django can reload the server if you change any scripts or templates.

We don't need to use shift + F10 to reload the server. Django will automatically update the data and reload the server. It saves time and makes it easier to handle the code.

# Admin

04

A default background management system in Django.

```
Terminal: Local x +

C:\Users\86159\PycharmProjects\ecofuler>python manage.py createsuperuser
Username (leave blank to use '86159'):
Email address:
Password:
Password (again):
The password is too similar to the username.
This password is entirely numeric.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.

C:\Users\86159\PycharmProjects\ecofuler>
```

Django administration

Username:  
86159

Password:  
.....

Log in

← → ↻ ⓘ 127.0.0.1:8000/admin/

应用 Gmail YouTube 地图

阅读清单

Django administration

WELCOME, 86159. [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

Site administration

AUTHENTICATION AND AUTHORIZATION

Groups

+ Add    ✎ Change

Users

+ Add    ✎ Change

Recent actions

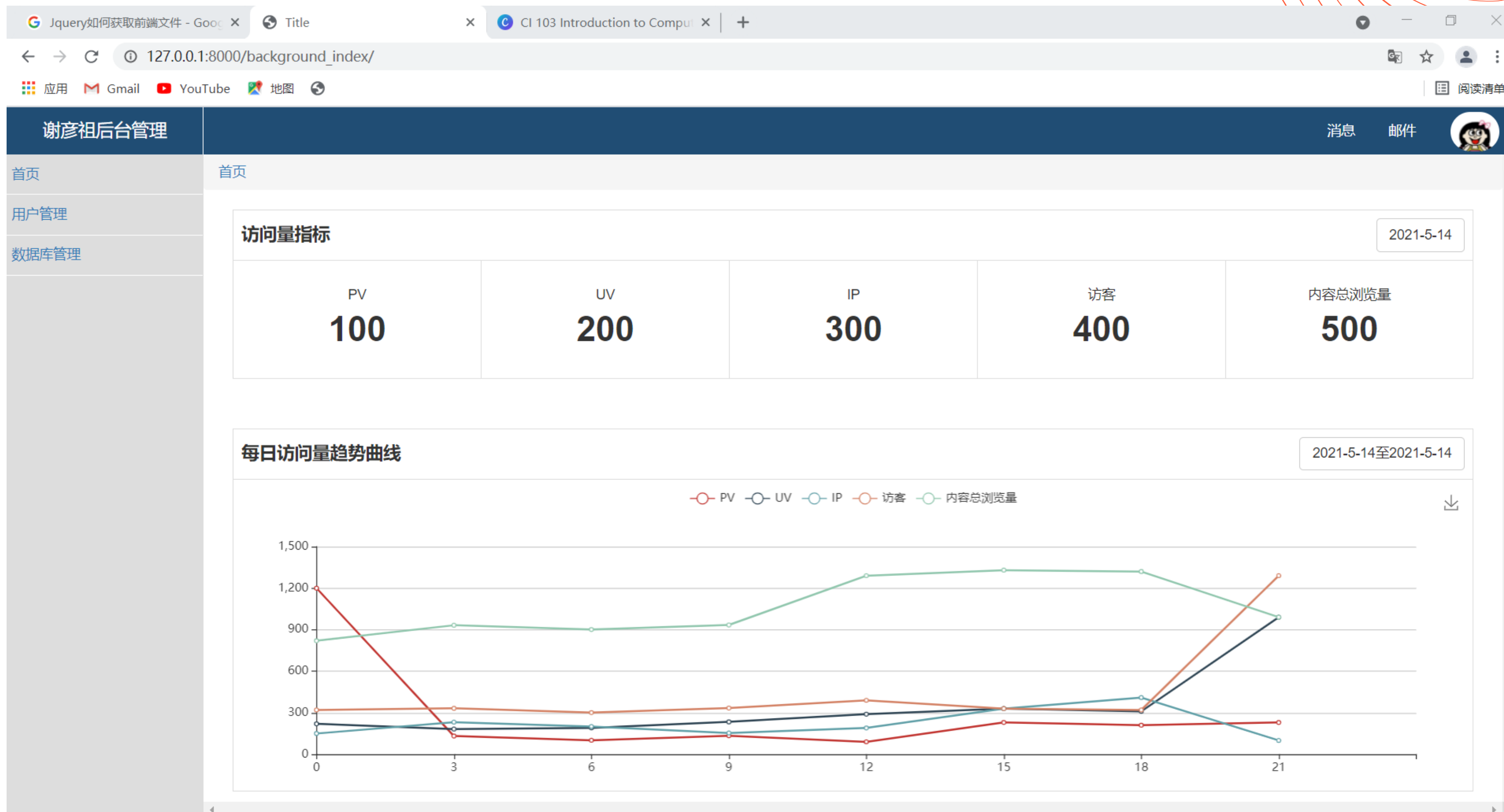
My actions

None available



# Back End View

04



The top left corner of the slide features a series of thin, light blue wavy lines that sweep across the frame, creating a sense of motion and depth.

05

# Technology Sharing 3: Database Design

# Data Processing

1. Balance sheet

2. Cash flow statement

3. Income statement

利润表（模板）					
编制单位：		年 月 日		金额单位：元	
项 目		本期金额		上期金额	
一、营业总收入					
减：	现金流量表（模板）				
编制单位：		年 月 日		单位：万元	
项 目		本期金额		上期金额	
一、经营活动产生的现金流量					
销售商品、提供劳务收到的现金					
收到的税费返还					
收到其他与经营活动有关的现金					
经营活动现金流入小计					
购买商品、接受劳务支付的现金					
支付给职工以及为职工支付的现金					
支付的各项税费					
支付其他与经营活动有关的现金					
经营活动现金流出小计					
经营活动产生的现金流量净额					
投资活动产生的现金流量					
收回投资收到的现金					
取得投资收益收到的现金					
处置固定资产、无形资产和其他长期资产收回的现金净额					
处置子公司及其他营业单位收到的现金净额					
收到其他与投资活动有关的现金					
投资活动现金流入小计					
购建固定资产、无形资产和其他长期资产支付的现金					
投资支付的现金					
取得子公司及其他营业单位支付的现金净额					
支付其他与投资活动有关的现金					
投资活动现金流出小计					
投资活动产生的现金流量净额					
筹资活动产生的现金流量					
吸收投资收到的现金					
发行债券收到的现金					
收到其他与筹资活动有关的现金					
筹资活动现金流入小计					
偿还债务支付的现金					
分配股利、利润或偿付利息支付的现金					
支付其他与筹资活动有关的现金					
筹资活动现金流出小计					
筹资活动产生的现金流量净额					
现金及现金等价物净增加额					
期初现金及现金等价物余额					
期末现金及现金等价物余额					

资产负债表（模板文件）					
公司名称：		年 月 日		单位：元	
资产	期末余额	年初余额	负债和所有者权益（或股东权益）	期末余额	年初余额
流动资产：			流动负债：		
货币资金			短期借款		
交易性金融资产			交易性金融负债		
应收票据			应付票据		
应收账款			应付账款		
预付款项			预付账款		
应收利息			应付职工薪酬		
二、投资			应交税金		
收回			应付利息		
取得			应付股利		
处置			其他应付款		
处置			一年内到期的非流动性负债		
收到			流动负债合计		
加：			非流动流动负债：		
减：			长期借款		
			应付债券		
			长期应付款		
			专项应付款		
三、利			预计负债		
减：			递延所得税负债		
			其他非流动负债		
			非流动负债合计		
			负债合计		
三、筹资			所有者权益（或股东权益）：		
吸收			实收资本（或股本）		
借款			资本公积		
收到			减：库存股		
筹			盈余公积		
			未分配利润		
			所有者权益（或股东权益）合计		
			负债和所有者权益（或股东权益）合计		

# Data Transforming

We consider changing the \*.xls into data into database. Python 3.9 allows us to do the work: This code allows us to read every line of excel.

```
import pymysql
import xlrd

# 连接数据库
try:
    db = pymysql.connect(host="127.0.0.1", user="root",
                        passwd="",
                        db="LzdxDB",
                        charset='utf8')
except:
    print("could not connect to mysql server")

def open_excel():
    try:
        book = xlrd.open_workbook('test.xls') # 文件名, 把文件与py文件放在同一目录下
    except:
        print("open excel file failed!")
    try:
        sheet = book.sheet_by_name("主要经济指标") # excel里面的worksheet1
        return sheet
    except:
        print("locate worksheet in excel failed!")

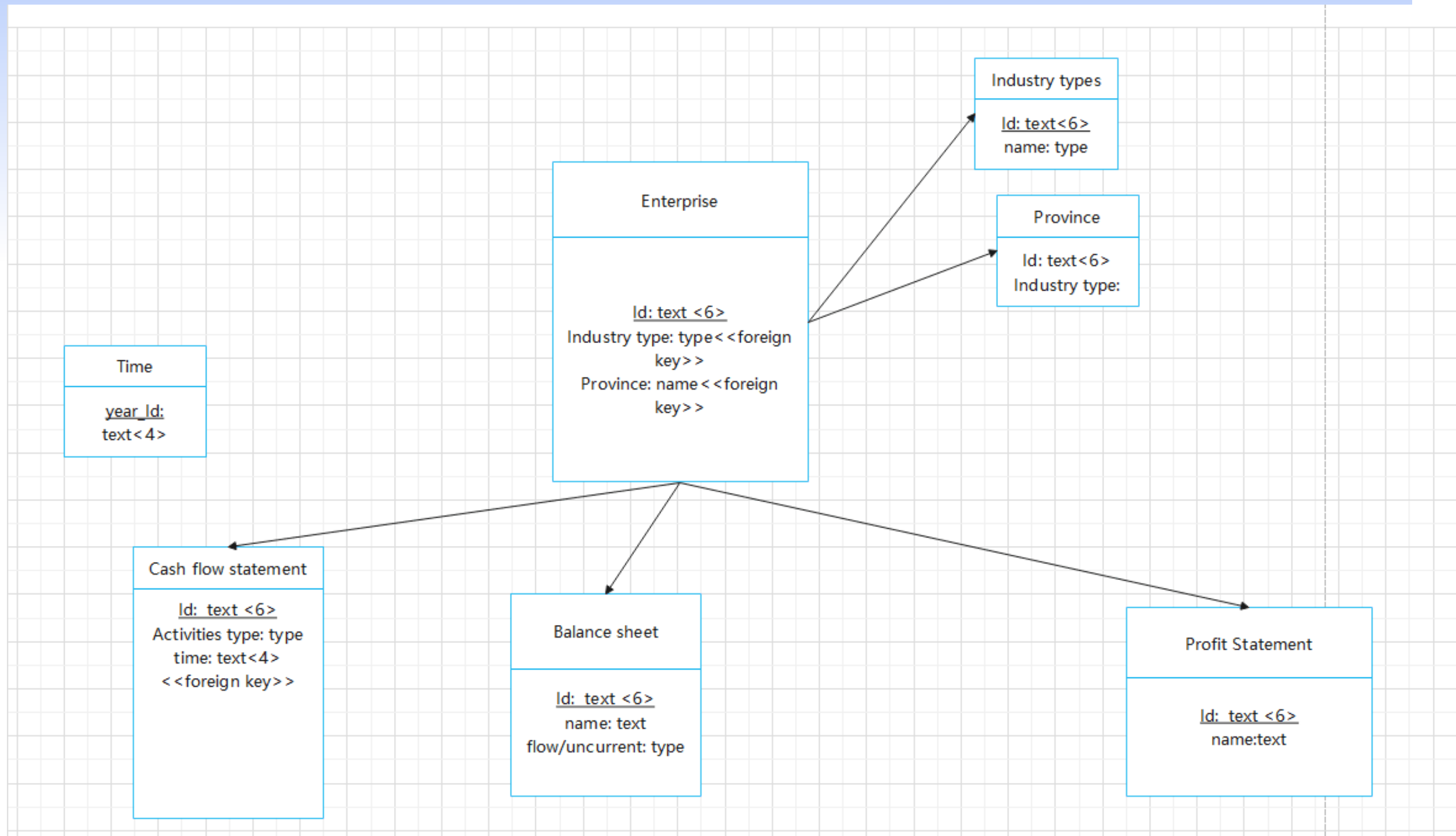
def insert_data():
    sheet = open_excel()
    cursor = db.cursor()
    row_num = sheet.nrows
    for i in range(1, row_num): # 第一行是标题名, 对应表中的字段名所以应该从第二行开始, 计算机以0开始计数, 所以值是1
        row_data = sheet.row_values(i)
        value = (row_data[0], row_data[1], row_data[2])
        print(i)
        sql = "INSERT INTO testTable(Name1 Name2 Name3)VALUES(%s %s %s)"
```



# Data Transforming Result

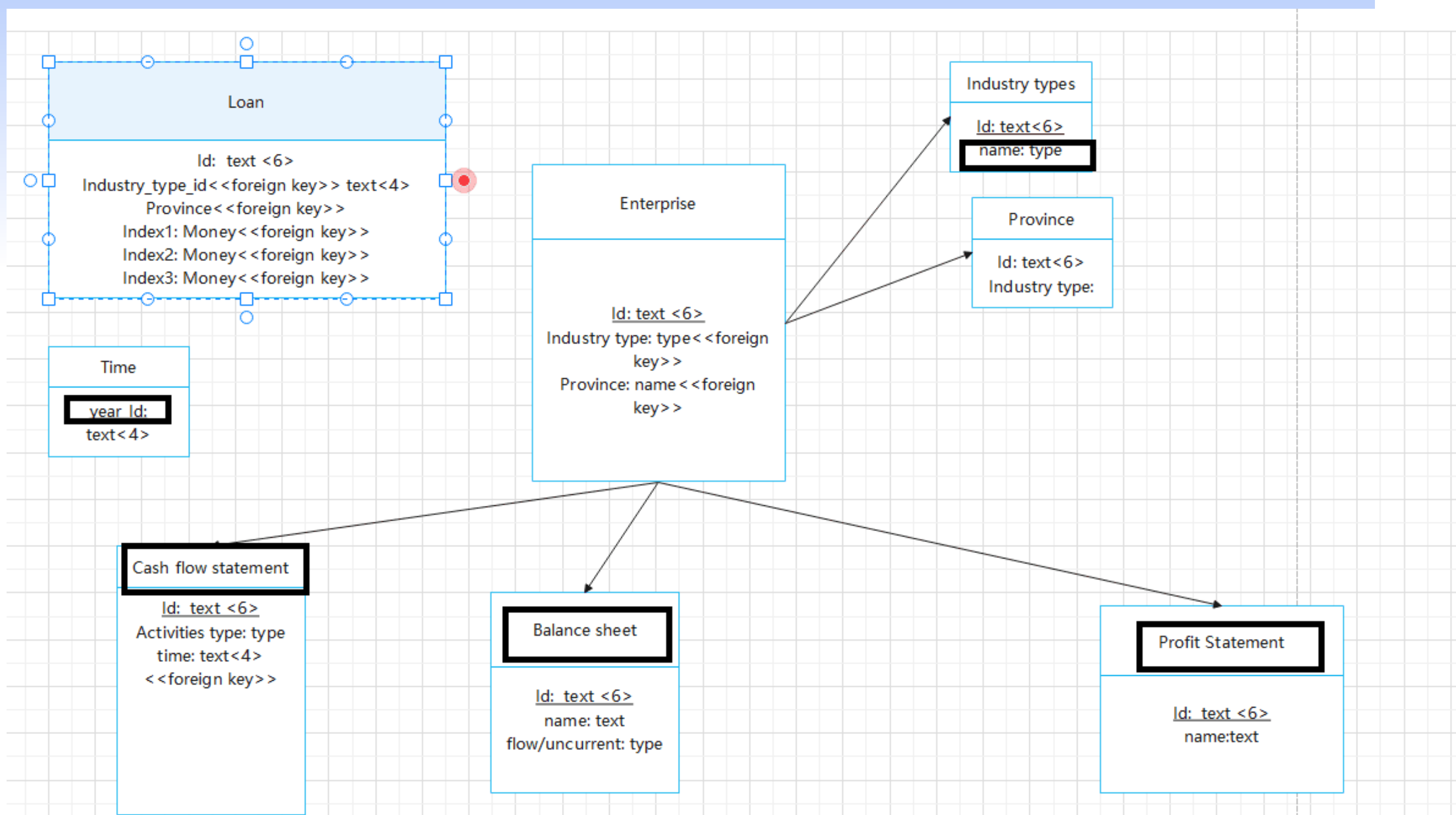
管理员: 命令提示符 - mysql -hlocalhost -uroot -p			
电力、热力生产和供应业	2012	4. 1. 6外商资本	
电力、热力生产和供应业	2012	5. 主营业务收入	
电力、热力生产和供应业	2012	6. 主营业务成本	
电力、热力生产和供应业	2012	7. 主营业务税金及附加	
电力、热力生产和供应业	2012	8. 销售费用	
电力、热力生产和供应业	2012	9. 管理费用	
电力、热力生产和供应业	2012	9. 1税金	
电力、热力生产和供应业	2012	10. 财务费用	
电力、热力生产和供应业	2012	10. 1利息收入	
电力、热力生产和供应业	2012	10. 2利息支出	
电力、热力生产和供应业	2012	11. 投资收益（损失以“-”号记）	
电力、热力生产和供应业	2012	12. 营业利润	
电力、热力生产和供应业	2012	13. 利润总额	
电力、热力生产和供应业	2012	14. 亏损企业亏损额	
电力、热力生产和供应业	2012	15. 应交增值税	
电力、热力生产和供应业	2012	16. 应交所得税	
燃气生产和供应业	2012	1. 工业销售产值（当年价格）	
燃气生产和供应业	2012	1. 1出口交货值	
燃气生产和供应业	2012	2. 资产总计	
燃气生产和供应业	2012	2. 1固定资产合计	
燃气生产和供应业	2012	2. 1. 1固定资产原价	
燃气生产和供应业	2012	2. 1. 2累计折旧	
燃气生产和供应业	2012	2. 2流动资产合计	
燃气生产和供应业	2012	2. 2. 1应收账款	
燃气生产和供应业	2012	2. 2. 1. 1存货	
燃气生产和供应业	2012	2. 2. 1. 1. 2产成品	
燃气生产和供应业	2012	3. 负债合计	
燃气生产和供应业	2012	3. 1流动负债合计	
燃气生产和供应业	2012	3. 1. 1应付账款	
燃气生产和供应业	2012	4. 所有者权益合计	

# Database Design (1): Data Schema

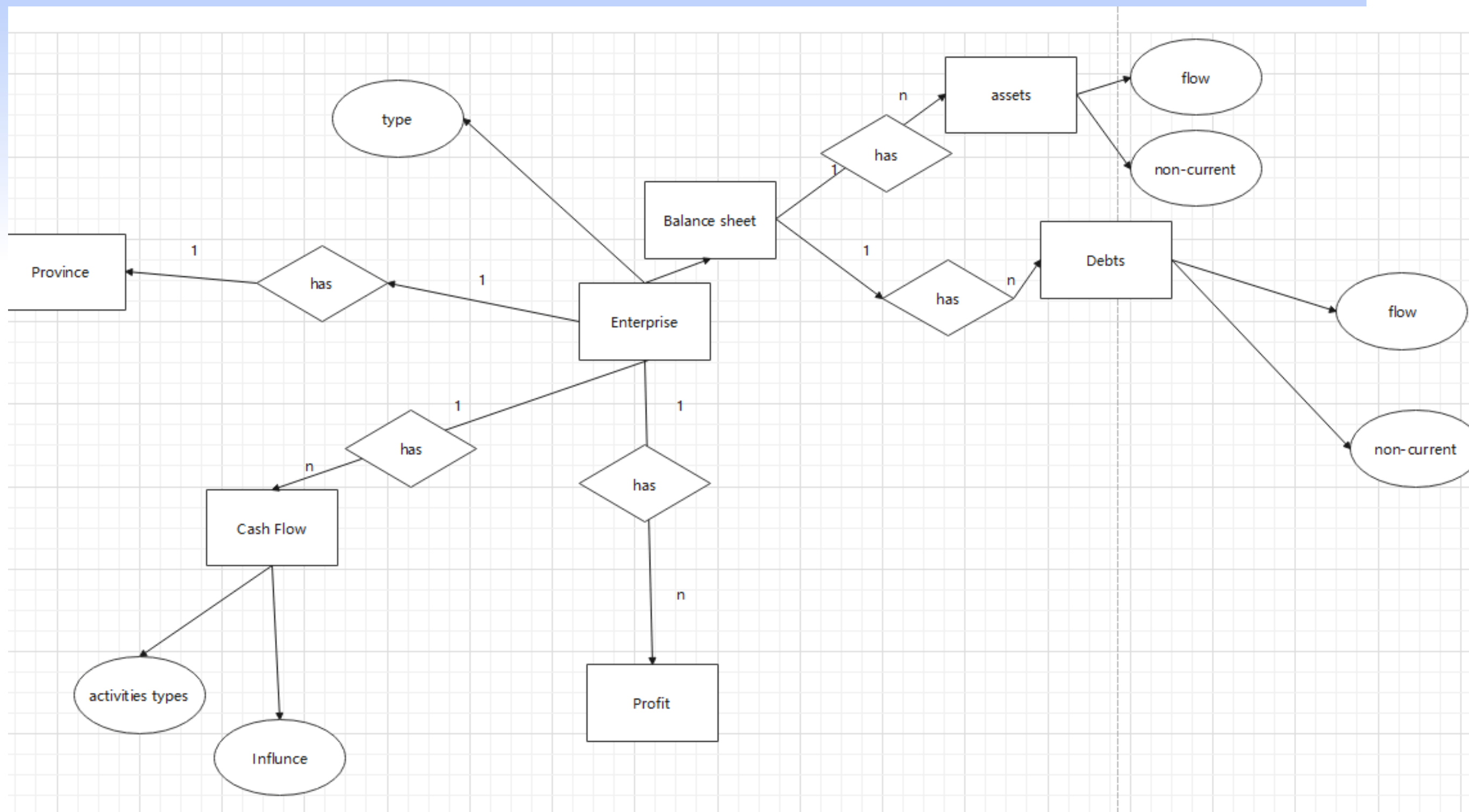




# Database Design (2): Joint Table



# Database Design (3): Entity-Relationship Graph



The background features a dark red, almost black, solid color. Overlaid on this are numerous thin, wavy, orange lines that flow across the frame in a fluid, organic pattern, resembling liquid or smoke. These lines are more densely packed in some areas and more spread out in others, creating a sense of movement and depth.

*Thank You for Listening*