

数据产品经理实战训练营

数据查询语言SQL

主讲：李硕



讲师：李硕 资深产品经理

曾就职于联想集团、乐视集团

擅长数据处理

擅长逻辑梳理

实战经验丰富

项目介绍

IOT：智能音箱、智能存储、空气净化器、智能路由器等

车联网项目：TSP系统、分时租赁系统、车机系统、客户管理系统等

C端：车主App、智能行车记录仪等

目录

C A T A L O G

- 数据库基础
- SQL查询基础
- 使用SQL进行数据分析

通过本次课

掌握：数据库及SQL基本概念

- (1) 数据库的分类

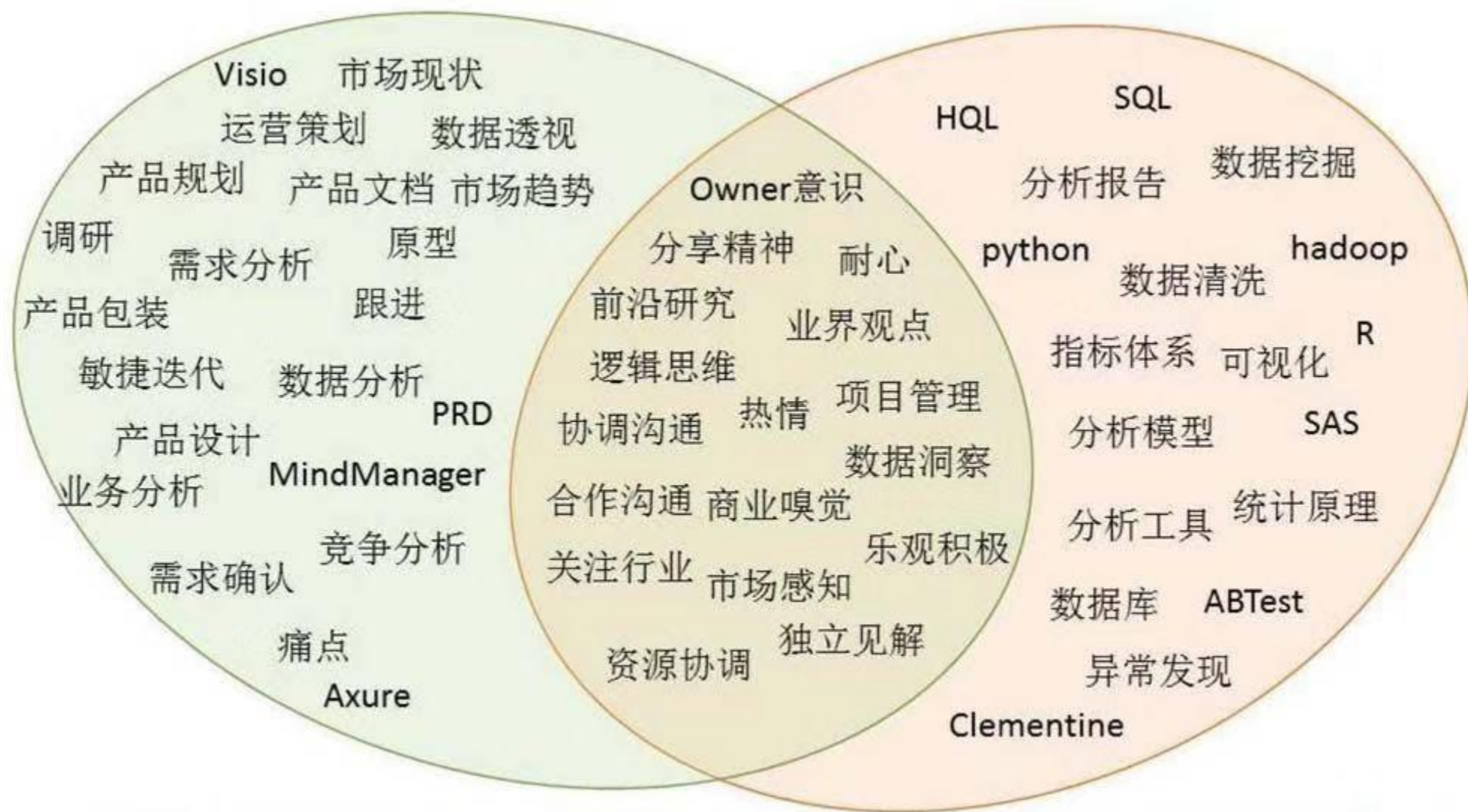
- (2) SQL的基本语法

应用：SQL的实际应用

- (1) 了解SQL在数据分析中的应用

- (2) 学会使用SQL实现数据分析

Part1：数据库基础



	滴滴	蚂蚁金服	vivo	爱财集团	曹操专车	综合
知识	1、本科以上学历 2、了解数据分析理念及工具 3、熟练掌握sql	1、掌握sql知识 2、了解hadoop hbase spark数据处理或存储平台	1、大学本科以上学历 2、熟悉sql		1、重点大学计算机、统计学等相关背景	1、本科及以上学历(3) 2、掌握sql知识(3)
硬技能	1、2年数据分析或数据产品经营 2、有hadoop/hive数据库设计经验 3、参与或者主导过大型数据体系经验 4、系统设计能力 5、数据分析能力	1、3年数据产品经验 2、具备数据挖掘能力 3、具备数据分析能力 4、有广告营销系统，搜索推荐系统，DMP标签系统的产品设计经验	1、优秀的数据分析能力 2、对效果执行数据进行总结归纳，并优化方案及策略 3、有项目管理能力 4、良好的时间管控能力 5、熟练使用office软件 6、熟练Axure软件	1、3年以上工作经验 2、具备需求调研能力 3、具备需求产品规划能力 4、具备产品设计能力 5、具体项目推进能力 6、大数据产品、数据分析、数据开发经验	1、4年以上数据相关产品相关工作经验 2、熟练掌握Hive、SQL 3、精通office 4、掌握python 5、有团队管理经验	1、3年及以上工作经验(3) 2、具备数据分析能力(3) 3、产品设计能力(3) 4、熟练使用Axure软件(2) 5、熟练使用office软件(2) 6、项目管理推进能力(2)
才干or软技能	1、业务抽象能力 2、数据敏感性 4、快速业务理解力 5、快速学习能力 6、逻辑能力强 7、商业直觉敏锐 8、用户体验嗅觉出色 9、良好的服务精神 10、良好的团队协作能力 11、良好的业务协调能力 12、抗压能力强	1、良好的沟通能力 2、良好的(团队)协作能力 3、良好的落地执行能力 4、有创新意识 5、有较强的跨部门推动及资源整合能力 6、始终保持工作激情和责任心	1、良好的沟通协调能力 2、较高的职业素养和敬业精神	1、较强的沟通能力 2、较强的产品管理能力 3、高效推动跨团队合作	1、对产品的基本方法论有结合实践经验的深刻认知 2、沟通表达能力佳 3、逻辑思维清晰，能够准确定位和剖析真实的业务需求	1、良好的沟通表达能力(4) 2、良好的团队协作、协调能力，跨团队合作能力(3) 3、较强的逻辑能力，逻辑思维清晰(2) 4、较强的产品管理能力(1) 5、出色的用户体验嗅觉(1)

I 从EXCEL说起

优点

简单易学

直观

方便更新、筛选及做简单的统计

缺点

无法支持复杂筛选

不方便做关联

不方便做统计

数据量大时响应缓慢

下单时间	取车时间	取车网点名称	取车long	取车lat	还车网点名称
2019/6/1 0:11	2019/6/1 0:11	天地软件园.金枫楼	121.4416	38.85668	玲珑居
2019/6/1 0:15	2019/6/1 0:15	天地软件园.金翔楼	121.4404	38.85902	高新园区福
2019/6/1 0:35	2019/6/1 0:35	斯坦福院落南门	121.5468	38.8806	大有文园东
2019/6/1 0:47	2019/6/1 0:47	数码广场软景E居网点	121.5406	38.8838	辽宁省农业
2019/6/1 0:50	2019/6/1 1:00	东北财经大学西门	121.5431	38.87779	玲珑居
2019/6/1 1:05	2019/6/1 1:05	亿达.峦翠园正门	121.5195	38.89028	海事大学人
2019/6/1 1:34	2019/6/1 1:36	数码广场软景E居网点	121.5406	38.8838	壹品漫谷西
2019/6/1 3:55	2019/6/1 3:55	东软河口园区F座地上	121.4994	38.85251	谷歌里
2019/6/1 4:07	2019/6/1 4:08	大有文园东门	121.5094	38.88442	大有文园东
2019/6/1 5:20	2019/6/1 5:21	星海公园宜客宜家酒店	121.5582	38.8765	星海公园宜
2019/6/1 5:32	2019/6/1 5:33	第七人民医院(西区)	121.516	38.87495	高新园区锦
2019/6/1 5:33	2019/6/1 5:37	大有文园东门	121.5094	38.88442	甘井子区金
2019/6/1 5:39	2019/6/1 5:40	百合山水IT家园	121.4947	38.88738	博苑大山幼
2019/6/1 5:59	2019/6/1 5:59	高新园区锦辉网点	121.5278	38.86422	数码广场软
2019/6/1 6:17	2019/6/1 6:19	新新园(东区)	121.5235	38.88751	高新园区锦
2019/6/1 6:18	2019/6/1 6:18	东北财经大学西门	121.5431	38.87779	星球花园
2019/6/1 6:23	2019/6/1 6:23	东北财经大学西门	121.5431	38.87779	东北财经大
2019/6/1 6:33	2019/6/1 6:34	BAC公司	121.5156	38.8663	BAC公司
2019/6/1 6:38	2019/6/1 6:38	大华锦绣华城3期西门	121.5167	38.89505	大华锦绣华
2019/6/1 6:46	2019/6/1 6:49	数码广场软景E居网点	121.5406	38.8838	高新区人才
2019/6/1 6:54	2019/6/1 7:04	高新区人才服务大厦	121.5113	38.84668	高新区人才
2019/6/1 7:06	2019/6/1 7:11	高新园区瑞丰园小区1	121.5134	38.84603	高新园区瑞
2019/6/1 7:07	2019/6/1 7:16	名仕乐居	121.5333	38.86922	甘井子区金
2019/6/1 7:07	2019/6/1 7:08	大华锦绣华城博雅园P	121.5167	38.89659	大华锦绣华
2019/6/1 7:12	2019/6/1 7:20	万科新里城	121.3957	38.84164	数码广场软
2019/6/1 7:14	2019/6/1 7:15	宅语原	121.5531	38.89484	宅语原
2019/6/1 7:14	2019/6/1 7:21	龙湖.大湖山语	121.51	38.87439	龙湖.大湖
2019/6/1 7:19	2019/6/1 7:19	万科新里城	121.3957	38.84164	大连启明法
2019/6/1 7:19	2019/6/1 7:19	高新园区瑞丰园小区E	121.5142	38.84624	高新园区瑞
2019/6/1 7:21	2019/6/1 7:22	宅语原	121.5531	38.89484	宅语原
2019/6/1 7:22	2019/6/1 7:22	数码广场软景E居网点	121.5406	38.8838	中海华庭西
2019/6/1 7:27	2019/6/1 7:30	美丽豪酒店	121.5194	38.85906	美丽豪酒店
2019/6/1 7:28	2019/6/1 7:28	宅语原	121.5531	38.89484	宅语原
2019/6/1 7:28	2019/6/1 7:29	智业广场	121.5138	38.85124	甘井子区金
2019/6/1 7:30	2019/6/1 7:31	博苑大山幼儿园	121.5009	38.8754	博苑大山幼
2019/6/1 7:30	2019/6/1 7:30	博苑大山幼儿园	121.5009	38.8754	博苑大山幼

了解数据库

按照一定的结构来组织、存储和管理数据的仓库。通常是由一个或一组文件组成的。



关系型数据库

键值对存储：Redis

文档存储：MongoDB

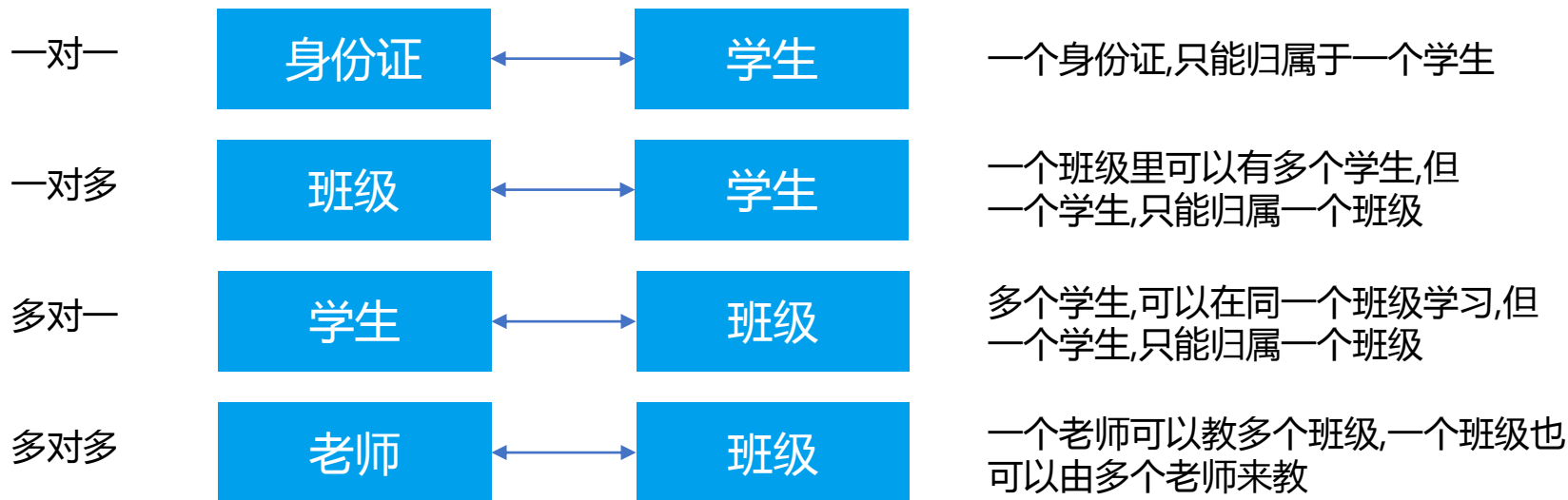
基于列的数据库：HBase

图形数据库：Neo4j

非关系型数据库

关系型数据库

关系模型



建立在**关系模型**基础上，由多张能互相联接的**二维行列表格**组成的数据库。

| 数据库基本结构

student_id	student_name	gender	birth_day	age	class_id	score	teacher_id
S20170015	张怡萍	男	2003-04-19	15	G0201	93.81	T0008
S20170016	周昊然	男	2004-07-01	14	G0201	47.86	T0008
S20170017	周义杰	男	2004-08-23	14	G0201	63.41	T0008
S20170018	方鸿晨	男	2006-10-09	12	G0202	90.16	T0009
S20170019	方逸	女	2005-01-10	13	G0202	60.56	T0009
S20170020	方曼	女	2003-09-13	15	G0202	59.72	T0009
S20170021	付晶灵	女	2003-07-22	15	G0202	79.87	T0009
S20170022	付君洁	女	2005-09-09	13	G0202	36.51	T0009
S20170023	付西	女	2004-10-28	14	G0202	72.50	T0009
S20170024	黄建钦	男	2006-02-21	12	G0202	55.23	T0009
S20170025	黄亦	男	2003-02-28	15	G0202	58.64	T0009
S20170026	李姍	女	2005-05-25	13	G0202	61.32	T0009
S20170027	李苑菲	女	2006-07-19	12	G0202	61.73	T0009
S20170028	任媛钰	女	2006-08-07	12	G0202	55.37	T0009
S20170029	田晨辰	男	2006-05-26	12	G0202	56.63	T0009

数据库

表

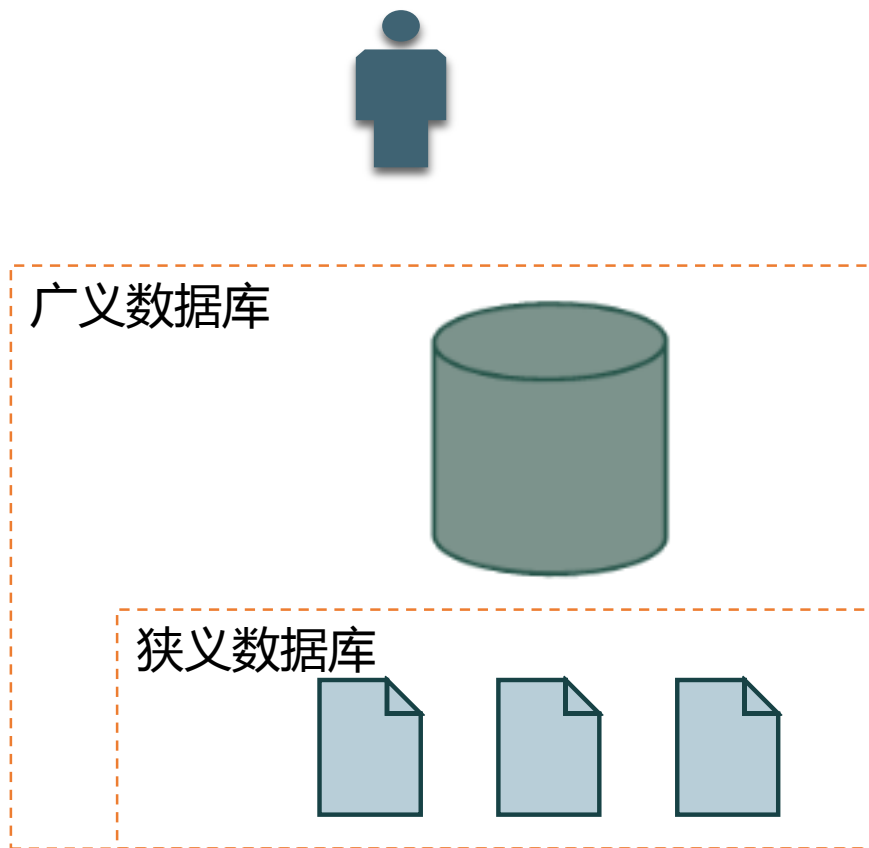
行、列

EXCEL文件

Sheet页

行（记录）
列（字段）

数据库基本结构

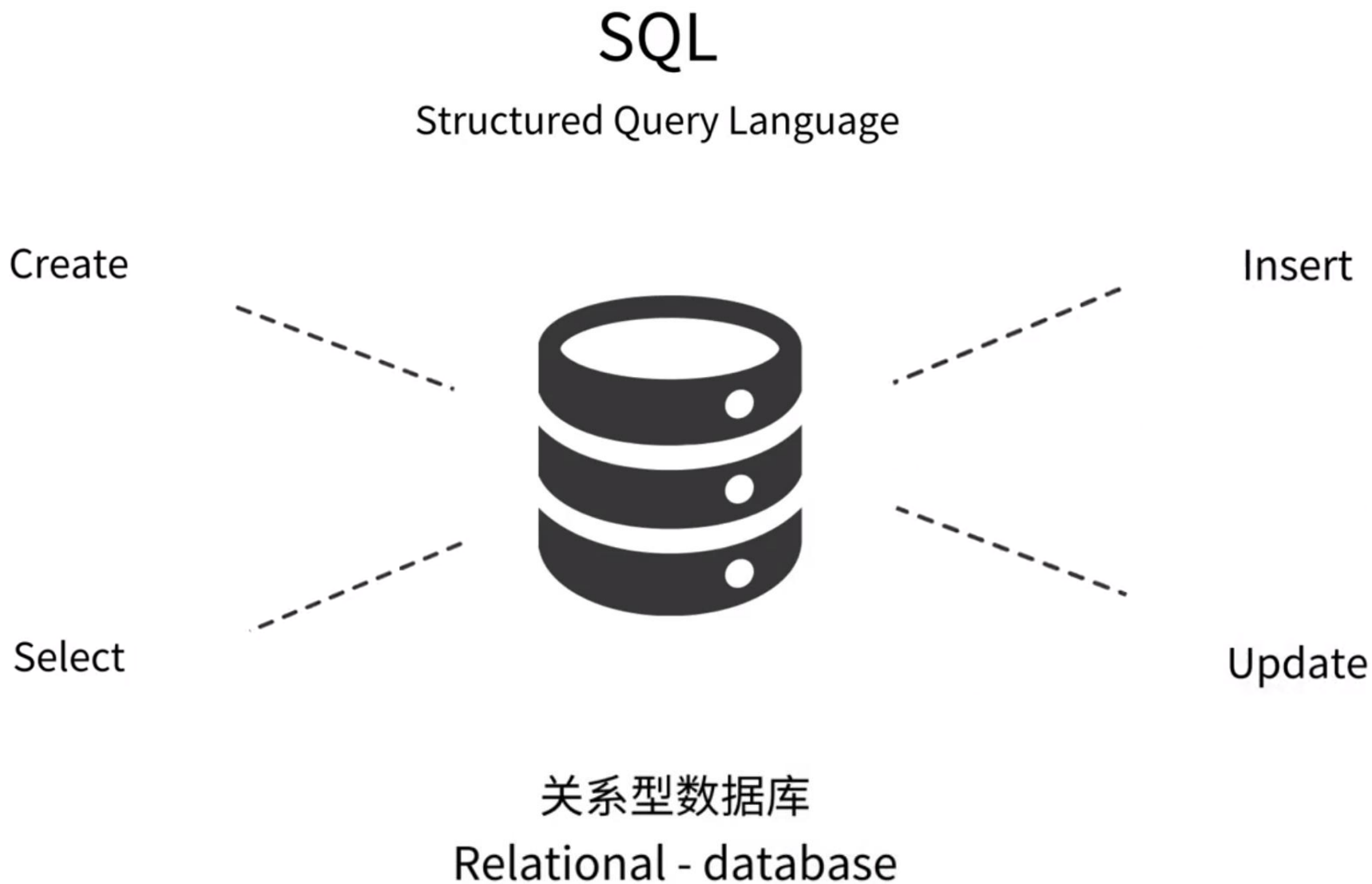


操作数据库：使用SQL语言

支持SQL操作

数据库管理系统（DBMS）：管理

一个或一组文件：组织、存储





官方下载地址：<https://www.mysql.com/downloads/>



官方下载地址：<https://www.navicat.com.cn/>



官方下载地址：<https://www.xp.cn/download.html>



了解数据库



关系型数据库



数据库基本结构



了解SQL



安装环境



MySQL介绍

Part2 : SQL查询基础

DDL

(Data Definition Language)

数据库定义语言：定义数据库的结构。

CREATE TABLE table_name

DROP DATABASE database_name

ALTER TABLE table_name

TRUNCATE TABLE table_name

DML

(Data Manipulation Language)

数据库操作语言：SQL中处理数据库中的数据

SELECT column FROM table_name UPDATE table_name

INSERT INTO table_name

DELETE FROM table_name

DCL

(Data Control Language)

数据库控制语言：授权，角色控制等

GRANT

REVOKE

TCL

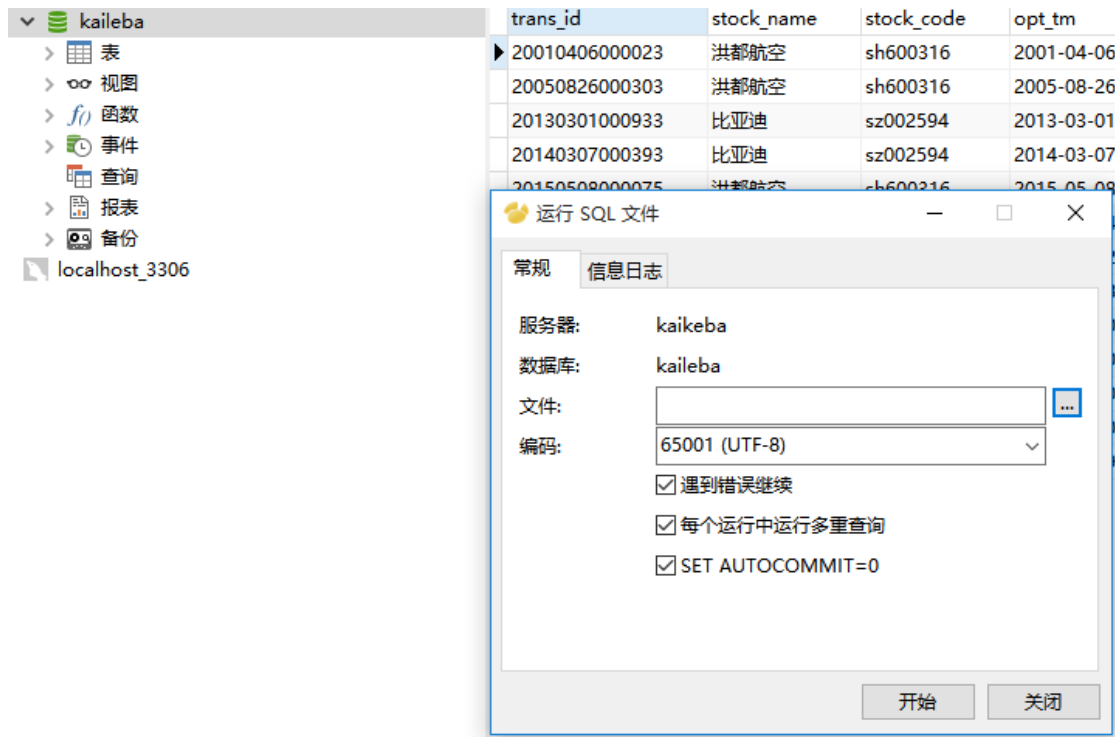
(Transaction Control Language)

事务控制语言

COMMIT

ROLLBACK

SAVEPOINT



查询部分列

查询数据：关键字SELECT

SELECT 字段列表 FROM 表名;

```
SELECT user_id FROM orderdata;
```

```
SELECT user_id,car_no,co_time FROM orderdata;
```

localhost	名称	类型	空	默...	属性	备注
information_schema	字段 (21)					
kaileba						
orderdata	od_time	varchar(255)	是	<空>		订单时间
od_time	co_time	varchar(255)	是	<空>		取车时间
co_time	pu_point	varchar(255)	是	<空>		取车网点名称
pu_point	pu_long	varchar(255)	是	<空>		取车经度
pu_long	pu_lat	varchar(255)	是	<空>		取车纬度
pu_lat	re_point	varchar(255)	是	<空>		还车网点名称
re_point	re_long	varchar(255)	是	<空>		还车经度
re_long	re_lat	varchar(255)	是	<空>		还车纬度
re_lat	mileage	varchar(255)	是	<空>		里程
mileage	duration	varchar(255)	是	<空>		时长
duration	user_id	varchar(255)	是	<空>		
user_id	order_id	varchar(255)	是	<空>		
order_id	car_no	varchar(255)	是	<空>		车牌号
car_no	car_type	varchar(255)	是	<空>		车型名称
car_type	package	varchar(255)	是	<空>		套餐名称
package	dri_amount	varchar(255)	是	<空>		行驶金额
dri_amount	payable_amount	varchar(255)	是	<空>		应付金额
payable_amount	disc_amount;	varchar(255)	是	<空>		优惠金额
disc_amount;	pay_amount	varchar(255)	是	<空>		实付金额
pay_amount	token	varchar(255)	是	<空>		代币支付
token	e	varchar(255)	是	<空>		
e						

查询所有列

SELECT * FROM orderdata;

kaileba

orderdata

od_time

co_time

pu_point

pu_long

pu_lat

re_point

re_long

re_lat

mileage

duration

user_id

order_id

car_no

car_type

package

dri_amount

payable_amount

disc_amount

pay_amount

token

e

student

student_id

student_name

gender

birth_day

age

class_id

score

teacher_id

t_stock_trans_dtl

teacher

3 SELECT * FROM orderdata;

4

X

Result 1

Result 2

Result 3

od_time	co_time	pu_point	pu_long	pu_lat	re_point	re_long	re_lat	mileage	duration	user_id	order_id	car_no	car_type	package	dri_amount	payable_amo...	disc_amount	pay_amount	token	e
2019/6/1 0:11	2019/6/1 0:11	天地软件园 金领楼	121.4415545	38.85667521	玲珑居	121.4404866	38.86434802	1	4	693757	2.02E+18	辽BD03621	一汽新特EV	标准定价	1.59	1.59	0	1.59	0	0至1
2019/6/1 0:15	2019/6/1 0:15	天地软件园 金领楼	121.4404137	38.85902198	高新区福莱园网	121.5100703	38.8549837	10	23	694141	2.02E+18	辽BD08522	一汽新特EV	标准定价	11.65	13.65	3.5	10.15	0	0至1
2019/6/1 0:35	2019/6/1 0:35	斯坦福笑落南	121.5468008	38.88060001	大有文园东门	121.5093723	38.88441801	5	13	643683	2.02E+18	辽BD02101	一汽新特EV	标准定价	6.2	6.2	1.86	4.34	0	0至1
2019/6/1 0:47	2019/6/1 0:47	数码广场秋景E座	121.5406009	38.88379936	辽宁省农业科学院	121.5196322	38.85607731	5	26	667898	2.02E+18	辽BD00498	长安逸动EV	标准定价	10.49	12.49	3.15	9.34	0	0至1
2019/6/1 0:50	2019/6/1 1:00	东北财经大学西门	121.543148	38.87779323	玲珑居	121.4404866	38.86434802	24	42	679750	2.02E+18	辽BD09806	长安逸动(香槟车)	标准定价	26.34	28.34	7.91	20.43	0	0至1
2019/6/1 1:05	2019/6/1 1:05	亿达·芯翠园正门	121.5194697	38.89027944	海事大学人才周转	121.5187561	38.87221175	4	17	683021	2.02E+18	辽BD00487	北汽EV150	标准定价	6.21	8.21	1.87	6.34	0	0至1
2019/6/1 1:34	2019/6/1 1:36	数码广场秋景E座	121.5406009	38.88379936	壹品漫谷西1门	121.4980846	38.8806139	26	246	656877	2.02E+18	辽BD00261	一汽新特EV	3小时套餐	69.04	77.04	31.52	45.52	1.52	4至6
2019/6/1 3:55	2019/6/1 3:55	东软河口园区F座1	121.499419	38.85250673	谷歌里	121.5051154	38.85254959	53	830	643741	2.02E+18	辽BD06400	海马爱尚EV	24小时套餐	89	97	28	69	0	6+
2019/6/1 4:07	2019/6/1 4:08	大有文园东门	121.5093723	38.88441801	大有文园东门	121.5093723	38.88441801	9	22	683184	2.02E+18	辽BD01119	一汽新特EV	标准定价	10.81	12.81	3.25	9.56	0	0至1
2019/6/1 5:20	2019/6/1 5:21	星海公园宜客宜家	121.5582328	38.87650322	星海公园宜客宜家	121.5582328	38.87650322	45	199	676361	2.02E+18	辽BD05544	海马爱尚EV	3小时套餐	54.86	62.86	10	52.86	0	4至6
2019/6/1 5:32	2019/6/1 5:33	第七人民医院(西	121.5159905	38.87494533	高新区锦辉网点	121.5278031	38.86421715	113	211	708513	2.02E+18	辽BD00405	长安逸动(香槟车)	3小时套餐	122.86	130.86	24	106.86	0	4至6
2019/6/1 5:33	2019/6/1 5:37	大有文园东门	121.5093723	38.88441801	甘井子区金家街网	121.5892026	38.96612501	15	34	10530	2.02E+18	辽BD01119	一汽新特EV	标准定价	17.35	19.35	5.21	14.14	0	0至1
2019/6/1 5:39	2019/6/1 5:40	百合山水汀家园	121.4946563	38.88737778	博苑大山幼儿园	121.5008922	38.87539524	1	11	705652	2.02E+18	辽BD02082	一汽新特EV	标准定价	3.34	5.34	1.76	3.58	0.75	0至1
2019/6/1 5:59	2019/6/1 5:59	高新区锦辉网点	121.5278031	38.86421715	数码广场秋景E座	121.5406009	38.88379936	3	13	677047	2.02E+18	辽BD03193	一汽新特EV	标准定价	5.02	7.02	1.51	5.51	0	0至1
2019/6/1 6:17	2019/6/1 6:19	新新园(东区)	121.5235185	38.88751271	高新区锦辉网点	121.5278031	38.86421715	4	15	652838	2.02E+18	辽BD00666	一汽新特EV	标准定价	6.11	6.11	0	6.11	0	0至1
2019/6/1 6:18	2019/6/1 6:18	东北财经大学西门	121.543148	38.87779323	星球花园	121.5330136	38.88111948	75	945	12812	2.02E+18	辽BD03813	一汽新特EV	24小时套餐	159	167	12	155	2	6+
2019/6/1 6:23	2019/6/1 6:23	东北财经大学西门	121.543148	38.87779323	东北财经大学西门	121.543148	38.87779323	9	45	42781	2.02E+18	辽BD06820	一汽新特EV	标准定价	16.56	18.56	0	18.56	0	0至1
2019/6/1 6:33	2019/6/1 6:34	BAC公司	121.5155578	38.86629901	BAC公司	121.5155578	38.86629901	66	795	2520	2.02E+18	辽BD05549	海马爱尚EV	24小时套餐	89	89	22.18	66.82	2.18	6+
2019/6/1 6:38	2019/6/1 6:38	大华锦绣华城三期	121.5167139	38.89504979	大华锦绣华城三期	121.5167139	38.89504979	46	166	695035	2.02E+18	辽BD07352	一汽新特EV	3小时套餐	64.34	70.34	0	70.34	0	1至3
2019/6/1 6:46	2019/6/1 6:49	数码广场秋景E座	121.5406009	38.88379936	高新区人才服务大	121.5113411	38.84668296	143	702	646964	2.02E+18	辽BD03193	一汽新特EV	24小时套餐	196.17	204.17	65	139.17	7	6+
2019/6/1 6:54	2019/6/1 7:04	高新区人才服务大	121.5113411	38.84668296	高新区人才服务大	121.5113411	38.84668296	51	111	8151	2.02E+18	辽BD01650	一汽新特EV	标准定价	57.84	61.84	5	56.84	0	1至3
2019/6/1 7:06	2019/6/1 7:11	高新区瑞丰园小	121.5133599	38.8460261	高新区瑞丰园小	121.5133599	38.8460261	34	78	33407	2.02E+18	辽BD00593	海马爱尚EV	标准定价	31.48	35.48	9.45	26.03	0	1至3
2019/6/1 7:07	2019/6/1 7:16	名仕东居	121.5332708	38.86922088	甘井子区金家街网	121.5892026	38.96612501	23	92	686897	2.02E+18	辽BD01712	一汽新特EV	标准定价	36.57	40.57	10	30.57	0	1至3
2019/6/1 7:07	2019/6/1 7:08	大华锦绣华城博雅	121.5166977	38.89659385	大华锦绣华城博雅	121.5166977	38.89659385	60	836	42327	2.02E+18	辽BD01826	一汽新特EV	24小时套餐	159	167	13.04	153.96	3.04	6+
2019/6/1 7:12	2019/6/1 7:20	万科新里城	121.3957127	38.84163971	数码广场秋景E座	121.5406009	38.88379936	19	43	698566	2.02E+18	辽BD00592	海马爱尚EV	标准定价	17.48	19.48	8.74	10.74	0	0至1
2019/6/1 7:14	2019/6/1 7:15	宅语原	121.5531443	38.89484445	宅语原	121.5531443	38.89484445	0	6	699772	2.02E+18	辽BD01195	一汽新特EV	6小时套餐	95	97	5	92	0	0至1
2019/6/1 7:14	2019/6/1 7:21	龙湖·大湖山语	121.510041	38.87439168	龙湖·大湖山语	121.510041	38.87439168	51	243	10663	2.02E+18	辽BD00846	海马爱尚EV	3小时套餐	66.16	74.16	10	64.16	0	4至6
2019/6/1 7:19	2019/6/1 7:19	万科新里城	121.3957127	38.84163971	大连启明海通信息	121.5070333	38.85718463	14	18	702290	2.02E+18	辽BD01293	一汽新特EV	标准定价	12.76	14.76	3.83	10.93	0	0至1
2019/6/1 7:19	2019/6/1 7:19	高新区瑞丰园小	121.5142297	38.84624021	高新区瑞丰园小	121.5153042	38.84617852	37	142	695961	2.02E+18	辽BD00719	一汽新特EV	3小时套餐	59.03	65.03	38.52	26.51	9	1至3

排序返回结果

将查询结果排序后展示：关键字ORDER BY ...
[ASC/DESC]

```
SELECT user_id,car_no,co_time,mileage  
FROM orderdata  
ORDER BY co_time ASC,mileage DESC
```

```
SELECT user_id,car_no,co_time FROM orderdata  
ORDER BY co_time DESC
```

4	SELECT user_id,car_no,co_time,mileage FROM orderdata ORDER BY co_time ASC,mileage DESC	
5	SELECT user_id,car_no,co_time FROM orderdata ORDER BY co_time DESC	
6		

×	user_id	car_no	co_time	mileage
	714095	辽BD08996	2019/10/1 0:00	1
	686160	辽BD07133	2019/10/1 0:01	2
	739748	辽BD02101	2019/10/1 0:07	6
	718276	辽BD00524	2019/10/1 0:10	74
	681034	辽BD01293	2019/10/1 0:14	14
	739732	辽BD01139	2019/10/1 0:15	10
	643683	辽BD01195	2019/10/1 0:19	5
	692057	辽BD01119	2019/10/1 0:26	6
	10941	辽BD06820	2019/10/1 0:30	27
	731526	辽BD07710	2019/10/1 0:38	13
	643683	辽BD09663	2019/10/1 0:40	9
	735873	辽BD00487	2019/10/1 0:41	14
	663803	辽BD03836	2019/10/1 0:57	16
	756696	辽BD02086	2019/10/1 10:00	118
	15499	辽BD03193	2019/10/1 10:03	11
	735278	辽BD09663	2019/10/1 10:04	4
	741855	辽BD00492	2019/10/1 10:06	6
	690632	辽BD00823	2019/10/1 10:14	4
	681461	辽BD00719	2019/10/1 10:14	3
	759698	辽BD03770	2019/10/1 10:24	24
	729163	辽BD00279	2019/10/1 10:29	4
	5000	辽BD00208	2019/10/1 10:30	27
	695070	辽BD06086	2019/10/1 10:30	10
	16550	辽BD08983	2019/10/1 10:32	63

限制返回结果

返回前几行：关键字LIMIT

- **SELECT 字段列表 FROM 表名 LIMIT 10;**

```
SELECT user_id,car_no,co_time  
FROM orderdata ORDER BY  
co_time DESC LIMIT 10
```

一般在两种情况下会使用：

- 想随便查看一下表中的数据；
- 一般与**ORDER BY**关键字配合使用，返回按某些字段排序后的前几行；

5	SELECT user_id,car_no,co_time FROM orderdata ORDER BY co_time DESC limit 10		
6			
x	user_id	car_no	co_time
	662199	辽BD01965	2019/9/9 9:59
	663477	辽BD08983	2019/9/9 9:58
	684089	辽BD00392	2019/9/9 9:58
	644023	辽BD00813	2019/9/9 9:56
	681130	辽BD02249	2019/9/9 9:55
	682987	辽BD00667	2019/9/9 9:54
	750379	辽BD08377	2019/9/9 9:54
	644338	辽BD00571	2019/9/9 9:51
	739173	辽BD03622	2019/9/9 9:50
	730955	辽BD03822	2019/9/9 9:49

限制返回结果

返回中间几行：关键字 **LIMIT M OFFSET N**（从第N行开始，返回M行记录）

两种写法：

- **SELECT 字段列表 FROM 表名 LIMIT M OFFSET N;**
- **SELECT 字段列表 FROM 表名 LIMIT N,M;**

```
SELECT user_id,car_no,co_time FROM  
orderdata ORDER BY co_time DESC  
LIMIT 10 OFFSET 2
```

```
SELECT user_id,car_no,co_time FROM  
orderdata ORDER BY co_time DESC LIMIT 2,10
```

4	SELECT	user_id,car_no,co_time,mileage	FROM orderdata ORDER BY co_time ASC,mileage DESC
5	SELECT	user_id,car_no,co_time	FROM orderdata ORDER BY co_time DESC limit 10 OFFSET 2
6			

×	user_id	car_no	co_time
	684089	辽BD00392	2019/9/9 9:58
	644023	辽BD00813	2019/9/9 9:56
	681130	辽BD02249	2019/9/9 9:55
	682987	辽BD00667	2019/9/9 9:54
	750379	辽BD08377	2019/9/9 9:54
	644338	辽BD00571	2019/9/9 9:51
	739173	辽BD03622	2019/9/9 9:50
	730955	辽BD03822	2019/9/9 9:49
	682996	辽BD02361	2019/9/9 9:47
	29787	辽BD09083	2019/9/9 9:46

限制返回结果

返回后几行：关键字LIMIT、ORDER BY

SQL语句中，没有返回后几行的专用写法，一般使用这种方式变相的返回后几行。

第1步：先按取车时间倒序排序；

第2步：返回排序后的前10行；

第3步：将前10行数据升序排序；

```
SELECT * FROM  
(  
  SELECT * FROM orderdata  
  ORDER BY co_time DESC LIMIT 10  
)ORDER BY co_time ASC
```

7 SELECT * FROM (SELECT * FROM orderdata ORDER BY co_time DESC LIMIT 10)a ORDER BY co_time ASC									
x	od_time	co_time	pu_point	pu_long	pu_lat	re_point	re_long	re_lat	mile
	2019/9/9 9:49	2019/9/9 9:49	海洋大学獐子岛教	121.5513221	38.86584786	海洋大学学生公寓	121.5377205	38.87213762	
	2019/9/9 9:50	2019/9/9 9:50	华南 华润凯旋门	121.5816955	38.99630612	西沟小区2区	121.550083	38.88233704	
	2019/9/9 9:51	2019/9/9 9:51	甘井子区金家街网	121.5892026	38.96612501	甘井子区金家街网	121.5892026	38.96612501	
	2019/9/9 9:54	2019/9/9 9:54	海洋大学獐子岛教	121.5513221	38.86584786	海洋大学獐子岛教	121.5513221	38.86584786	
	2019/9/9 9:49	2019/9/9 9:54	福荣园小区	121.5100703	38.8549837	星海公园宜客宜家	121.5582328	38.87650322	
	2019/9/9 9:52	2019/9/9 9:55	小平岛悦丽海湾小	121.5047683	38.84097594	龙海大厦	121.512849	38.8485649	
	2019/9/9 9:56	2019/9/9 9:56	数码广场软景E居F	121.5406009	38.88379936	数码广场软景E居F	121.5406009	38.88379936	
	2019/9/9 9:58	2019/9/9 9:58	大连市第四中学	121.5529208	38.88557343	大连市第四中学	121.5529208	38.88557343	
	2019/9/9 9:58	2019/9/9 9:58	数码广场软景E居F	121.5406009	38.88379936	大连市轻工业学校	121.5357191	38.87398997	
	2019/9/9 9:59	2019/9/9 9:59	小平岛悦丽海湾小	121.5047683	38.84097594	东软河口园区F座1	121.499419	38.85250673	

使用WHERE子句

过滤数据：关键字WHERE

SELECT 字段列表 FROM 表名 WHERE 过滤条件;

过滤条件一般由

要过滤的字段、操作符、限定值三部分组成；

如：

```
SELECT user_id,car_no,car_type FROM orderdata  
WHERE car_type = '海马爱尚EV';
```

SELECT user_id,car_no,car_type FROM orderdata WHERE car_type = '海马爱尚EV';		
user_id	car_no	car_type
643741	辽BD06400	海马爱尚EV
676361	辽BD05544	海马爱尚EV
2520	辽BD05549	海马爱尚EV
33407	辽BD00593	海马爱尚EV
698566	辽BD00592	海马爱尚EV
10663	辽BD00846	海马爱尚EV
641732	辽BD01002	海马爱尚EV
22973	辽BD00524	海马爱尚EV
673405	辽BD00844	海马爱尚EV
704754	辽BD00944	海马爱尚EV
651782	辽BD00554	海马爱尚EV
692500	辽BD01722	海马爱尚EV
822	辽BD01086	海马爱尚EV
686537	辽BD00545	海马爱尚EV
701432	辽BD04115	海马爱尚EV
652139	辽BD00592	海马爱尚EV
706351	辽BD00895	海马爱尚EV
709466	辽BD00554	海马爱尚EV
643541	辽BD01086	海马爱尚EV
684935	辽BD00905	海马爱尚EV
709370	辽BD00592	海马爱尚EV
706012	辽BD02258	海马爱尚EV
22973	辽BD00524	海马爱尚EV
691918	辽BD00540	海马爱尚EV
693757	辽BD00938	海马爱尚EV
645235	辽BD00513	海马爱尚EV
640675	辽BD00538	海马爱尚EV
643275	辽BD00813	海马爱尚EV
679720	辽BD01046	海马爱尚EV

常用操作符

分类	操作符	说明	适用范围
过滤单个值	=	等于	字符型、数值型、日期和时间型
	<>、!=	不等于	
	>	大于	
	>=	大于等于	
	!>	不大于	
	<	小于	
	<=	小于等于	
	!<	不小于	
过滤NULL值	IS NULL	为NULL值	
	IS NOT NULL	不为NULL值	
过滤集合	BETWEEN	在指定的两个值之间	仅适用于字符型
	IN	包含	
	NOT IN	不包含	
	EXISTS	存在于	
	NOT EXISTS	不存在于	
	LIKE	匹配LIKE关键字后的模式	
	NOT LIKE	不匹配LIKE关键字后的模式	

| 过滤单个值

```
SELECT user_id,car_no,car_type FROM orderdata WHERE car_type = '海马爱尚EV';  
SELECT user_id,car_no,mileage FROM orderdata WHERE mileage > 100;  
SELECT user_id,car_no,pu_point FROM orderdata WHERE pu_point = '天地软件园.金枫楼';
```

| 过滤NULL值

```
SELECT user_id,car_no,token FROM orderdata WHERE token is NULL;  
SELECT user_id,car_no,token FROM orderdata WHERE token is not NULL;  
SELECT user_id,car_no,token FROM orderdata WHERE token = NULL; ( 错误写法! )
```

| 过滤集合

```
SELECT user_id,car_no,pay_amount FROM orderdata WHERE pay_amount  
BETWEEN 10 AND 15;
```

```
SELECT user_id,car_no,co_time FROM orderdata WHERE co_time  
BETWEEN '2019-08-01 00:00:00' AND '2019-09-01 00:00:00';
```

```
SELECT user_id,car_no,pu_point FROM orderdata WHERE pu_point  
IN ('天地软件园.金枫楼','BAC公司');
```

```
SELECT user_id,car_no,car_type FROM orderdata WHERE car_type  
NOT IN ('海马爱尚EV','一汽新特EV');
```

使用通配符过滤数据

使用通配符过滤数据：关键字LIKE

百分号 %	匹配0~多个任意字符
下划线 _	匹配1个任意字符

```
SELECT user_id,user_name,car_type FROM orderdata  
WHERE user_name LIKE '李%';
```

```
SELECT user_id,user_name,car_type FROM orderdata  
WHERE user_name LIKE '%飞';
```

```
SELECT user_id,user_name,car_type FROM orderdata  
WHERE user_name LIKE '李_轩';
```

```
SELECT user_id,user_name,car_type FROM orderdata WHERE user_name LIKE '李%';  
SELECT user_id,user_name,car_type FROM orderdata WHERE user_name LIKE '%飞';  
SELECT user_id,user_name,car_type FROM orderdata WHERE user_name LIKE '李_轩';
```

user_id	user_name	car_type
704754	李定轩	一汽新特EV
700700	李定轩	一汽新特EV
38181	李定轩	一汽新特EV
703868	李定轩	长安逸动(香味车)
664839	李定轩	一汽新特EV
10110	李定轩	海马爱尚EV
644459	李定轩	长安逸动(香味车)
703699	李定轩	一汽新特EV
694692	李定轩	海马爱尚EV
705564	李定轩	一汽新特EV
701763	李定轩	一汽新特EV
40145	李定轩	一汽新特EV
33758	李定轩	海马爱尚EV
682958	李定轩	一汽新特EV
691747	李定轩	海马爱尚EV
706266	李定轩	一汽新特EV
6144	李定轩	海马爱尚EV
701269	李定轩	海马爱尚EV
11942	李定轩	一汽新特EV
12071	李定轩	海马爱尚EV
679424	李定轩	海马爱尚EV
660447	李定轩	一汽新特EV
38874	李定轩	海马爱尚EV

使用逻辑操作符组合WHERE子句：

AND操作符	满足所有条件
OR操作符	满足任一条件

```
SELECT user_id,user_name,car_type FROM orderdata
WHERE user_name LIKE '李%' AND car_type = '海马爱尚EV';
SELECT user_id,user_name,car_type FROM orderdata
WHERE user_name LIKE '李%' OR car_type = '海马爱尚EV';
```

```
SELECT user_id,user_name,car_type,mileage FROM orderdata
WHERE mileage > 100
AND (user_name LIKE '李%' OR car_type = '海马爱尚EV');
```

使用组合WHERE子句时，尽量都使用括号消除歧义

```
SELECT user_id,user_name,car_type,mileage FROM orderdata
WHERE mileage > 100 AND
(user_name LIKE '李%' OR car_type = '海马爱尚EV');
```

user_id	user_name	car_type	mileage
669902	李姗	一汽新特EV	145
709466	周喻 h	海马爱尚EV	104
709414	王琳	海马爱尚EV	106
703852	方逸	海马爱尚EV	124
710215	吴少雄	海马爱尚EV	129
710099	李咏颐	一汽新特EV	101
703939	吴岩滨	海马爱尚EV	103
707313	吴少雄	海马爱尚EV	106
661181	汪涵颖	海马爱尚EV	112
14484	李欣	北汽EV150	143
16913	张蓝一	海马爱尚EV	129
12595	胡昊泽	海马爱尚EV	113
711543	李舒飞	一汽新特EV	151
660848	吴锦	海马爱尚EV	217
706266	李定轩	一汽新特EV	107
649196	牛苑	海马爱尚EV	170



SQL的分类



基础查询数据



排序返回结果



过滤返回结果

Part3：使用SQL进行数据分析

子查询：嵌套在其他查询中的查询

如何同时查询出学生编号、学生姓名、老师编号、老师姓名？

```
SELECT
  student_id,
  student_name,
  teacher_id,
  (
    SELECT teacher_name
    FROM teacher
    WHERE teacher.teacher_id = student.teacher_id
  )
FROM student;
```

子查询使用表名消除歧义

```
SELECT
  student_id,
  student_name,
  teacher_id,
  (
    SELECT teacher_name
    FROM teacher
    WHERE teacher.teacher_id = student.teacher_id
  )
FROM student;
```

student_id	student_name	teacher_id	(SELECT t...
S20180001	方东美	T0003	李一萱
S20180002	方香	<NULL>	<NULL>
S20180003	高紫茵	T0003	李一萱
S20180004	胡未迟	T0003	李一萱
S20180005	李咏颐	T0003	李一萱
S20180006	吴灏潇	<NULL>	<NULL>
S20180007	吴明鸿	T0003	李一萱
S20180008	吴鹏宇	T0003	李一萱
S20180009	吴少雄	T0003	李一萱
S20180010	习芬飘	T0003	李一萱
S20180011	俞倚琳	T0003	李一萱
S20180012	张琼雪	T0003	李一萱
S20180013	陈顺军	T0004	刘金霞
S20180014	方浩杰	<NULL>	<NULL>
S20180015	方静雅	T0004	刘金霞
S20180016	胡博涵	<NULL>	<NULL>
S20180017	胡昊泽	T0004	刘金霞
S20180018	牛思静	T0004	刘金霞
S20180019	刘佳乐	T0004	刘金霞
S20180020	刘笑婷	T0004	刘金霞
S20180021	梅慧瑶	T0004	刘金霞

使用子查询过滤数据

获取姓刘的老师教的年龄大于12岁的学生信息

```
SELECT
student_id,student_name,age,teacher_id,
( SELECT teacher_name
  FROM teacher
  WHERE teacher.teacher_id = student.teacher_id
)FROM
(
  SELECT *
  FROM student
  WHERE student.age >= 12
)student
WHERE teacher_id
IN ( SELECT teacher_id FROM teacher
    WHERE teacher_name like '刘%');
```

```
SELECT
student_id,student_name,age,teacher_id,
(
  SELECT teacher_name
  FROM teacher
  WHERE teacher.teacher_id = student.teacher_id
)
FROM
(
  SELECT *
  FROM student
  WHERE student.age >= 12
)student
WHERE teacher_id
IN ( SELECT teacher_id FROM teacher
    WHERE teacher_name like '刘%');
```

student_id	student_name	age	teacher_id	(SELECT t...
S20180013	陈顺军	12	T0004	刘金霞
S20180017	胡昊泽	12	T0004	刘金霞
S20180018	牛思静	12	T0004	刘金霞
S20180020	刘笑婷	12	T0004	刘金霞
S20180022	牛佳艳	12	T0004	刘金霞
S20180023	牛芸	12	T0004	刘金霞
S20180024	任美梦	12	T0004	刘金霞
S20180025	汪涵颖	12	T0004	刘金霞
S20180027	吴昭溶	12	T0004	刘金霞
S20180032	高齐瑞	12	T0005	刘思哲
S20180033	黄建国	12	T0005	刘思哲
S20180037	牛灵雪	12	T0005	刘思哲
S20180038	牛紫萍	12	T0005	刘思哲
S20180039	任美梦	12	T0005	刘思哲

使用子查询过滤数据 (EXISTS)

```
SELECT
student_id, student_name, age, teacher_id,
(
    SELECT teacher_name
    FROM teacher
    WHERE teacher.teacher_id = student.teacher_id
)
FROM
(
    SELECT *
    FROM student
    WHERE student.age >= 12
) student
WHERE EXISTS
( SELECT * FROM teacher
  WHERE student.teacher_id = teacher.teacher_id AND teacher.teacher_name like '刘%' );
```

student_id	student_name	age	teacher_id	(SELECT t...
320180013	陈顺军	12	T0004	刘金霞
320180017	胡昊泽	12	T0004	刘金霞
320180018	牛思静	12	T0004	刘金霞
320180020	刘笑婷	12	T0004	刘金霞
320180022	牛佳艳	12	T0004	刘金霞
320180023	牛芸	12	T0004	刘金霞
320180024	任美梦	12	T0004	刘金霞
320180025	汪涵颖	12	T0004	刘金霞
320180027	吴昭溶	12	T0004	刘金霞
320180032	高齐濡	12	T0005	刘思哲
320180033	黄建国	12	T0005	刘思哲
320180037	牛灵雪	12	T0005	刘思哲
320180038	牛紫莹	12	T0005	刘思哲

```
66 SELECT
67 student_id, student_name, age, teacher_id,
68 (
69     SELECT teacher_name
70     FROM teacher
71     WHERE teacher.teacher_id = student.teacher_id
72 )
73 FROM
74 (
75     SELECT *
76     FROM student
77     WHERE student.age >= 12
78 ) student
79 WHERE NOT EXISTS
80 ( SELECT * FROM teacher
81   WHERE student.teacher_id = teacher.teacher_id AND teacher.teacher_name like '刘%' );
82
```

student_id	student_name	age	teacher_id	(SELECT t...
S20180001	方东美	12	T0003	李一萱
S20180003	高紫茵	12	T0003	李一萱
S20180010	习芬飘	13	T0003	李一萱
S20180012	张琼雪	12	T0003	李一萱
S20170001	陈慧	14	T0008	梅艺涵
S20170002	方松阳	13	T0008	梅艺涵
S20170003	方以晴	14	T0008	梅艺涵
S20170004	方悦倩	14	T0008	梅艺涵
S20170005	付朗丽	14	T0008	梅艺涵
S20170006	胡耀幻	14	T0008	梅艺涵
S20170007	胡宇诚	12	T0008	梅艺涵
S20170008	黄路	15	T0008	梅艺涵
S20170009	李姗	15	T0008	梅艺涵
S20170010	梅芝	14	T0008	梅艺涵

UNION ALL 与 UNION

相同点：都是用来合并多个结果集

不同点：UNION ALL合并结果集后不去除重复记录；
UNION合并结果集后去除重复记录；

使用组合查询合并结果集

合并来源于不同的表的结果集

```
SELECT * FROM student WHERE age = 10  
UNION  
SELECT * FROM student WHERE class_id = 'G0101'  
UNION  
SELECT * FROM student WHERE gender = '男';
```

```
SELECT student_id,student_name FROM student  
UNION  
SELECT teacher_id,teacher_name FROM teacher;
```

```
SELECT student_id,student_name,age FROM student  
UNION  
SELECT teacher_id,teacher_name,gender FROM teacher;
```

虽然不会报错，但也要注意的几点：

- 合并后的结果集的title与第一个结果集保持一致。
- 待合并的结果集的字段的顺序、数据类型及字段值的含义尽量保持一致。

总共有多少名学生？	<code>SELECT COUNT(*) FROM student;</code>
所有学生的数学成绩总和？	<code>SELECT SUM(score) FROM student;</code>
所有学生的数学成绩最高分？	<code>SELECT MAX(score) FROM student;</code>
所有学生的数学成绩最低分？	<code>SELECT MIN(score) FROM student;</code>
所有学生的数学成绩平均分？	<code>SELECT AVG(score) FROM student;</code>

COUNT(*)	SUM(score)	MAX(score)	MIN(score)	AVG(score)
219	14419.93	99.45	10.57	65.844429

如何获取每个班级的成绩平均分（因为学生太多，这里先不计算成绩在80分以下的学生），以便比较不同班级的成绩？

使用关键字：**GROUP BY**

```
SELECT  
class_id, avg(score)  
FROM student  
WHERE score > 80  
GROUP BY class_id;
```

class_id	score
G0101	80.65
G0101	88.84
G0101	84.69
G0101	83.42
G0101	97.38
G0101	81.01
G0102	91.13
G0103	80.76
G0103	85.71
G0103	94.7
G0103	85.08
G0104	82.69
G0104	81.31
G0104	97.02
G0104	80.89
G0105	91.85
G0105	98.99
G0105	84.38

class_id	score
G0101	80.65
G0101	88.84
G0101	84.69
G0101	83.42
G0101	97.38
G0101	81.01

class_id	score
G0102	91.13

class_id	score
G0103	80.76
G0103	85.71
G0103	94.7
G0103	85.08

class_id	score
G0104	82.69
G0104	81.31
G0104	97.02
G0104	80.89

class_id	score
G0105	91.85
G0105	98.99
G0105	84.38

class_id	avg(score)
G0101	85.99833
G0102	91.13
G0103	86.5625
G0104	85.4775
G0105	91.74

比如：如何获取教授15名以上学名的老师？

使用关键字：GROUP BY + HAVING

```
SELECT teacher_id,count(*) FROM student GROUP BY teacher_id HAVING count(*) > 15;
```

比如：如何获取每个班级的数学成绩平均分（不计算成绩在80分以下的学生且过滤掉平均分在90分以下的班级），以便比较不同班级的成绩？

使用关键字：GROUP BY + HAVING

```
SELECT  
    class_id,avg(score)  
FROM student WHERE score > 80 GROUP BY class_id HAVING avg(score) > 90;
```

WHERE

过滤行

在GROUP BY前执行

区别？

HAVING

过滤分组

在GROUP BY后执行

子句	作用	是否必须使用
SELECT	指定要返回的字段或表达式	是
FROM	指定检索数据来源	否。只在从表中取数时必须，其他情况下可不使用
WHERE	过滤行	否
GROUP BY	指定分组字段或表达式	否。只在分组计算使用聚集函数时必须，其他情况下不使用
HAVING	过滤分组	否
ORDER BY	指定排序字段或表达式	否

获取每个班级的数学成绩平均分（不计算成绩在80分以下的学生且过滤掉平均分在90分以下的班级），然后以平均分从高到低排序？

```
SELECT
    class_id,avg(score)
FROM student
WHERE score > 80
GROUP BY class_id
HAVING avg(score) > 90
ORDER BY avg(score) DESC;
```

```
106 SELECT
107     class_id,avg(score)
108 FROM student
109 WHERE score > 80
110 GROUP BY class_id
111 HAVING avg(score) > 90
112 ORDER BY avg(score) DESC;
113
114
115
```

x	class_id	avg(score)
	G0105	91.740000
	G0303	91.676667
	G0201	91.322000
	G0102	91.130000

常见函数分类

函数分类	示例
数值函数	类型转换函数、绝对值函数、指数函数。。。
文本函数	拼接函数、截取函数、转换函数。。。
日期和时间函数	格式化函数、截取函数、系统日期。。。

常见数值函数

函数	作用	示例
abs(n)	返回n的绝对值	abs(-3) = 3
round(n,d)	返回n的四舍五入值，保留d位小数	round(3.234,2) = 3.23
rand()	返回0~1之间的随机数	rand() 可能返回0.2511382673
pow(x,y)	返回x的y次幂	pow(2,3) = 8
mod(m,n)	返回m除以n的余数	mod(7,3) = 1
三角函数	实现三角运算的函数。如sin,cos,tan等	sin(90) = 0.89399666360055785

- 返回一个随机数
- 返回2的3次方
- 从订单表中返回实付费用（只保留1位小数位）

```
14 SELECT rand();
```

```
rand()
```

```
0.6505219728225116
```

```
115 SELECT pow(2,3);
```

```
× pow(2,3)
```

```
8
```

```
116 SELECT pay_amount, round(pay_amount,1) FROM orderdata;
```

pay_amount	round(pay_amount,1)
1.59	1.6
10.15	10.2
4.34	4.3
9.34	9.3
20.43	20.4
6.34	6.3
45.52	45.5
69	69.0
9.56	9.6
52.86	52.9
106.86	106.9
14.14	14.1
3.58	3.6
5.51	5.5
6.11	6.1
155	155.0

常见文本函数

函数	作用	示例
length(str)	返回字符串str的长度	length('ABCDE') = 5
left(str,n)	返回字符串str的左端n个字符	left('ABCDE' ,3) = 'ABC'
right(str,n)	返回字符串str的右端n个字符	right('ABCDE' ,3) = 'CDE'
trim(str)	删除str左右两边的空格	trim(' ABC ')= 'ABC'
upper(str)	将字符串str转为大写	upper('Abc') = 'ABC'
lower(str)	将字符串str转为小写	lower('Abc') = 'abc'
substring(str,m,n)	返回从字符串str第m位后面的n个字符	substring('ABCDE' ,2,3) = 'BCD'
instr(str1,str2)	返回字符串str2在str1中首次出现的位置	instr('foobarbar', 'bar') = 4
concat(str1,str2,...)	返回字符串str1,str2,...按顺序拼接后的结果	concat('ABC','DEFG','H') = 'ABCDEFGH'

常见日期和时间函数

函数	作用	示例
current_date()	返回当前日期	current_date()可能的返回值' 2018-02-27'
current_time()	返回当前时间	current_time()可能的返回值' 12:34:10'
year(date)	返回日期date所在的年份	year('2018-02-27') = 2018
month(date)	返回日期date所在的月份	month('2018-02-27') = 2
day(date)	返回日期date所在的日	day('2018-02-27') = 27
hour(date)	返回日期date所在的小时	hour(' 12:40:32') = 12
minute(date)	返回日期date所在的分钟	minute(' 12:40:32') = 40
dayofyear(date)	返回date是当年的第几天	dayofyear(' 2018-02-27') = 58
dayofmonth(date)	返回date是当月的第几天	dayofmonth(' 2018-02-27') = 27
dayofweek(date)	返回date是当周的第几天 (第1天是周日)	dayofweek(' 2018-02-27') = 3
date_format(date, 格式)	返回date格式化后的结果	date_format(current_date(),'%Y') = 2018

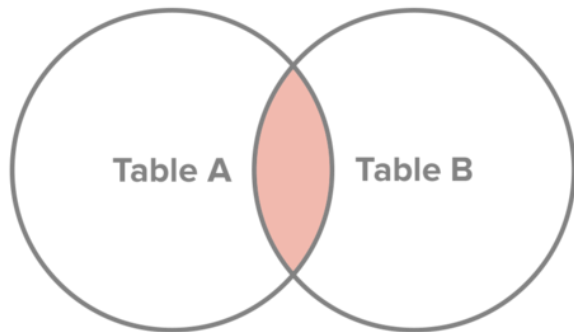
返回订单表中订单时间和取车时间日期（只返回年月日，去掉时分秒）

```
SELECT od_time,date_format(od_time,'%Y-%m-%d'),  
co_time,date_format(co_time,'%Y-%m-%d')  
FROM orderdata;
```

```
118 SELECT od_time,date_format(od_time,'%Y-%m-%d'),co_time,date_format(co_time,'%Y-%m-%d')
119 FROM orderdata;
120
121
...
```

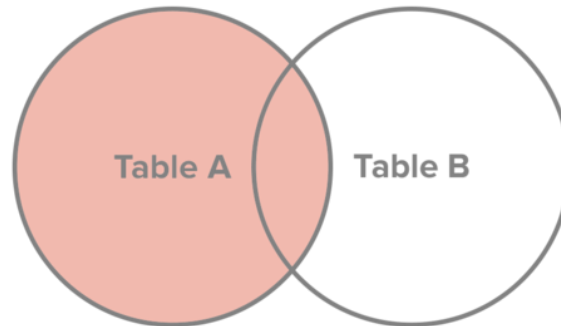
x	od_time	date_format(o...	co_time	date_format(c...
	2019-06-01 00:11:00	2019-06-01	2019-06-01 00:11:00	2019-06-01
	2019-06-01 00:15:00	2019-06-01	2019-06-01 00:15:00	2019-06-01
	2019-06-01 00:35:00	2019-06-01	2019-06-01 00:35:00	2019-06-01
	2019-06-01 00:47:00	2019-06-01	2019-06-01 00:47:00	2019-06-01
	2019-06-01 00:50:00	2019-06-01	2019-06-01 01:00:00	2019-06-01
	2019-06-01 01:05:00	2019-06-01	2019-06-01 01:05:00	2019-06-01
	2019-06-01 01:34:00	2019-06-01	2019-06-01 01:36:00	2019-06-01
	2019-06-01 03:55:00	2019-06-01	2019-06-01 03:55:00	2019-06-01
	2019-06-01 04:07:00	2019-06-01	2019-06-01 04:08:00	2019-06-01
	2019-06-01 05:20:00	2019-06-01	2019-06-01 05:21:00	2019-06-01
	2019-06-01 05:32:00	2019-06-01	2019-06-01 05:33:00	2019-06-01
	2019-06-01 05:33:00	2019-06-01	2019-06-01 05:37:00	2019-06-01
	2019-06-01 05:39:00	2019-06-01	2019-06-01 05:40:00	2019-06-01
	2019-06-01 05:59:00	2019-06-01	2019-06-01 05:59:00	2019-06-01
	2019-06-01 06:17:00	2019-06-01	2019-06-01 06:19:00	2019-06-01
	2019-06-01 06:18:00	2019-06-01	2019-06-01 06:18:00	2019-06-01

Inner Join



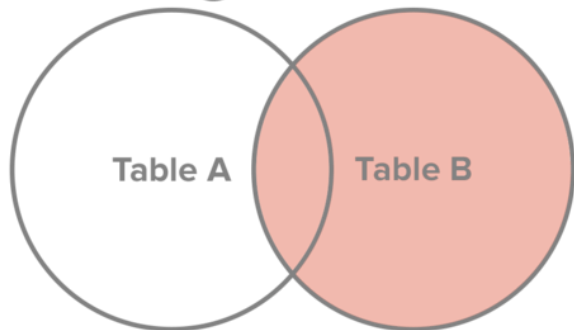
内连接 (inner join)

Left Join



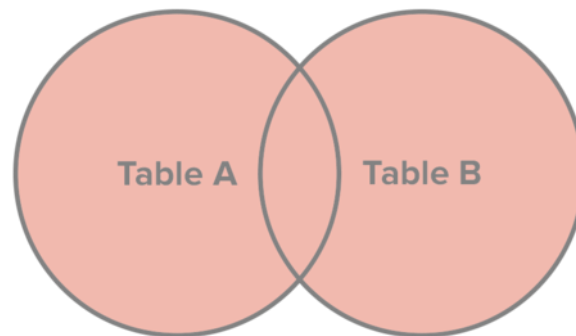
左连接 (left join)

Right Join

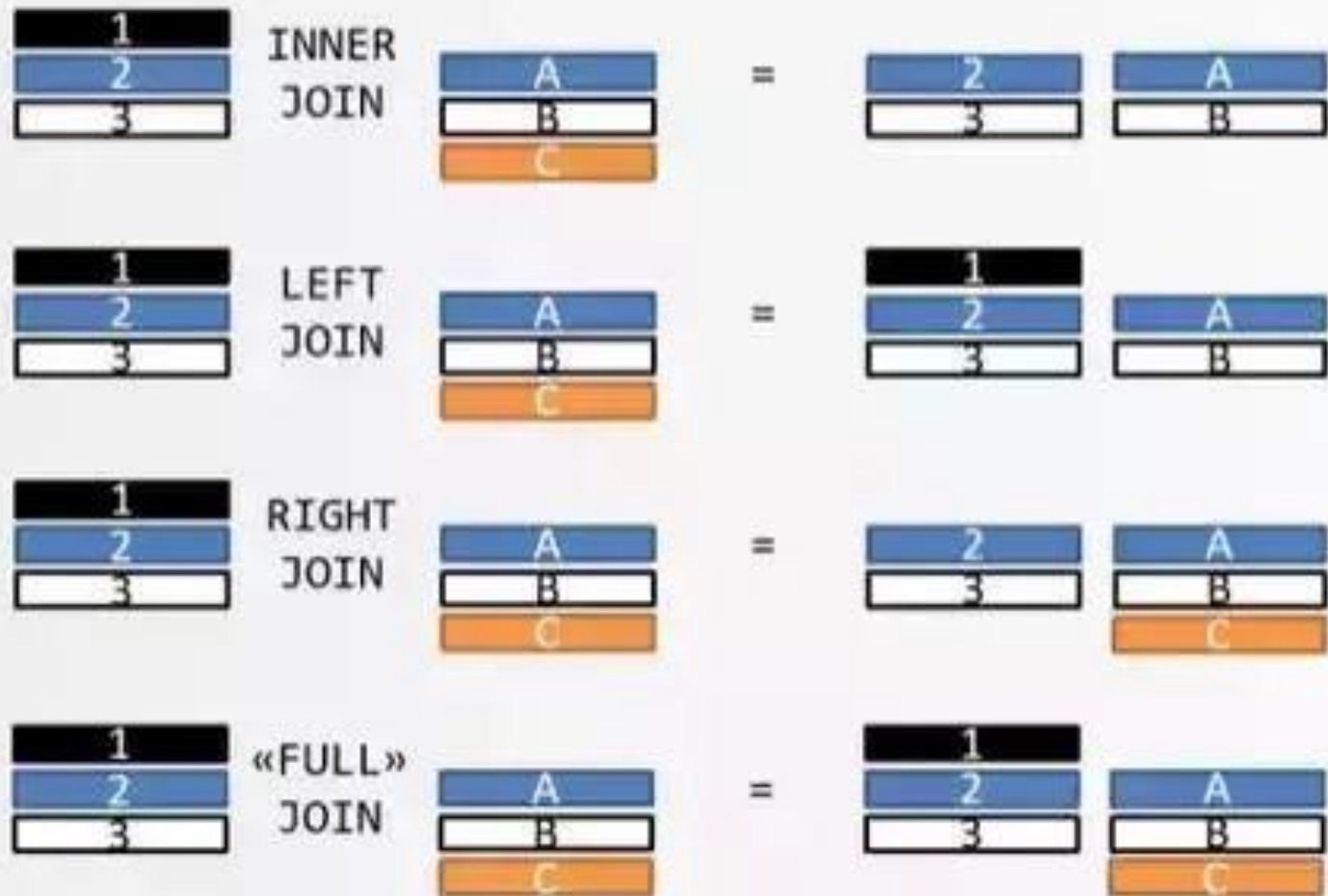


右连接 (right join)

Full Join



全连接 (full join)



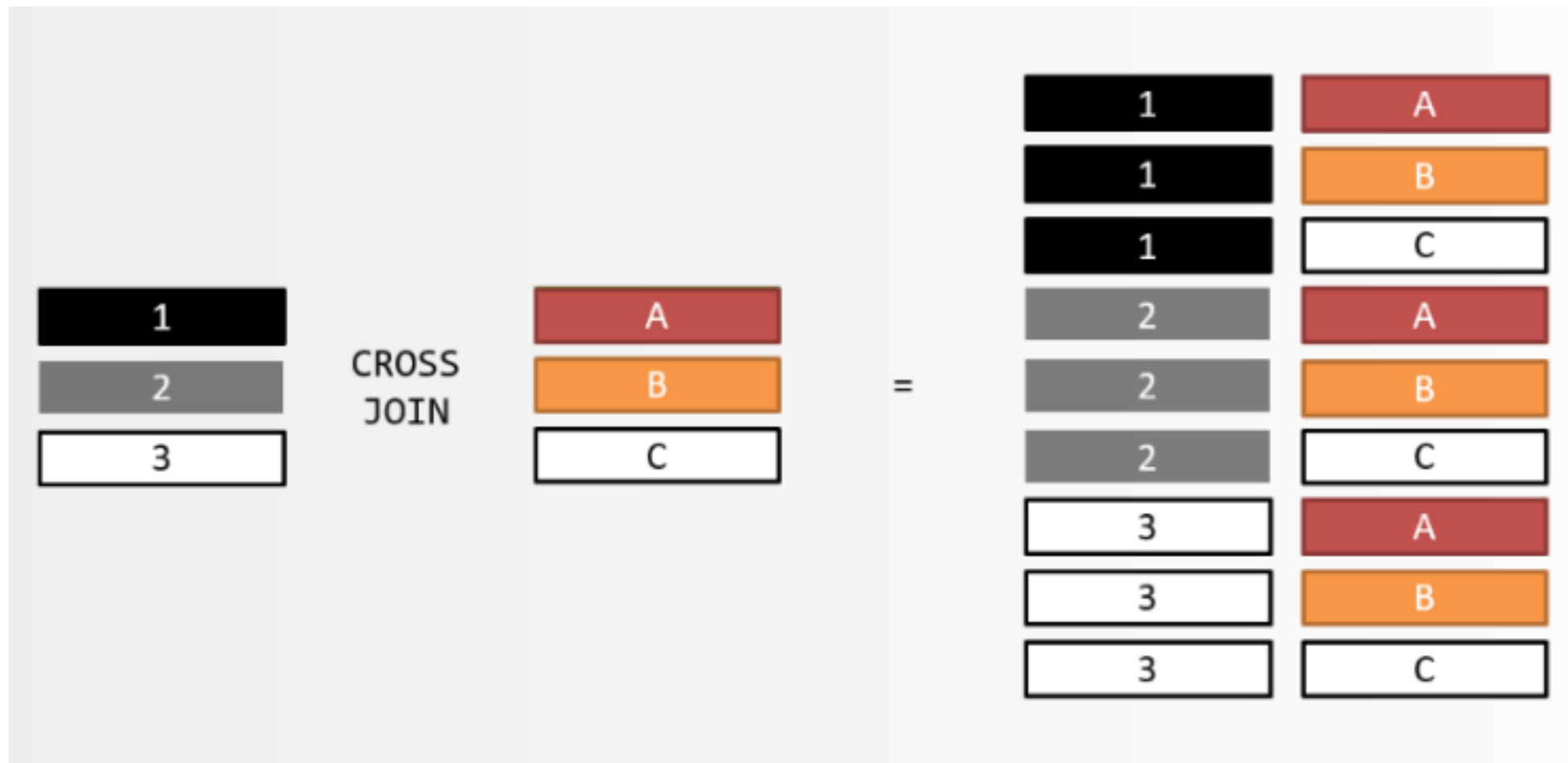
这四种连接的 SQL 语句如下。

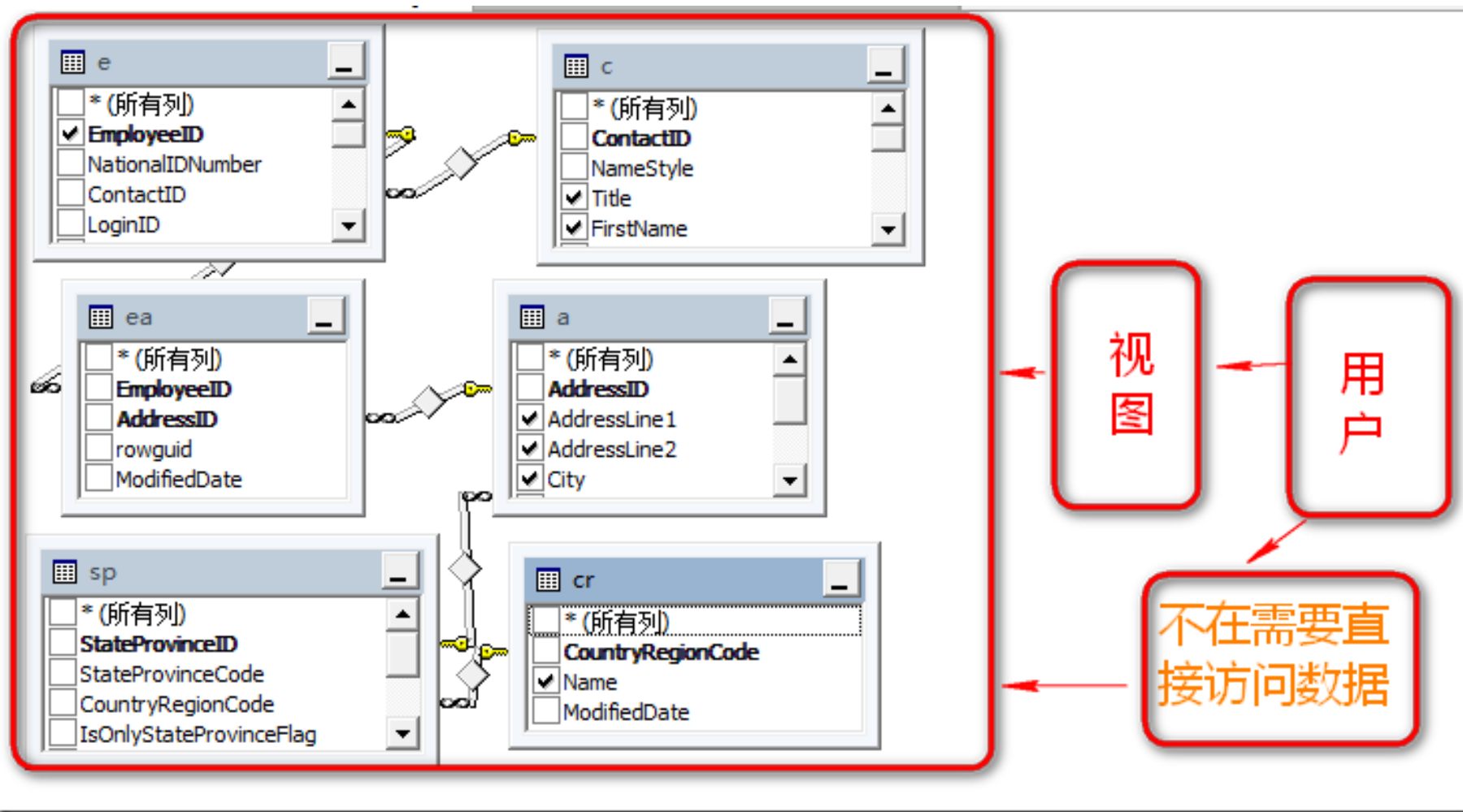
```
SELECT * from student  
INNER JOIN teacher ON student.teacher_id = teacher.teacher_id;
```

```
SELECT * from student  
LEFT JOIN teacher ON student.teacher_id = teacher.teacher_id;
```

```
SELECT * from student  
RIGHT JOIN teacher ON student.teacher_id = teacher.teacher_id;
```

```
SELECT * from student  
FULL JOIN teacher ON student.teacher_id = teacher.teacher_id;
```



创建视图：使用**CREATE VIEW**关键字。语法如下：

```
CREATE VIEW 视图名  
AS  
SELECT子句;
```

比如，查询学生信息时，同时查询出老师姓名：

```
CREATE VIEW v_student  
AS  
SELECT  
  a.*,b.teacher_name  
FROM student a  
LEFT JOIN teacher b  
ON a.teacher_id = b.teacher_id;
```

查询视图：与查询表一样，使用SELECT子句。

比如，查询学生信息时，同时查询出老师姓名：

```
SELECT * FROM v_student;  
SELECT student_id,student_name,teacher_id,teacher_name from v_student;  
SELECT * FROM v_student WHERE teacher_id = 'T0003';
```

视图的嵌套：查询所有考试及格的学生信息：

```
CREATE VIEW v_student_nesting  
AS  
SELECT * FROM v_student  
WHERE score >= 60;
```

表

存储数据

不依赖于其他表或视图

增、删、改、查

视图

不存储数据，所以不占用物理存储

来源于表及其他视图

查，不适合增、删、改

常见的使用场景

场景一：仅提供需要的数据；

场景二：对特定的用户仅开放特定的数据，达到保护敏感数据的目的，提升了数据安全性；

场景三：仅筛选需要的数据

场景四：简化复杂的操作

场景五：重新格式化出新的字段

场景六：使用计算表达式生成新的字段

场景七：屏蔽底层实现逻辑及频繁的变更

场景八：合并多个分离的子表



过滤数据及函数处理



汇总统计



子查询



组合查询



连接表



视图



课程总结

重点：

数据库及SQL基本概念
SQL的基本语法

难点：

SQL在数据分析中的应用
使用SQL实现数据分析

Thanks

数据产品经理实战训练营