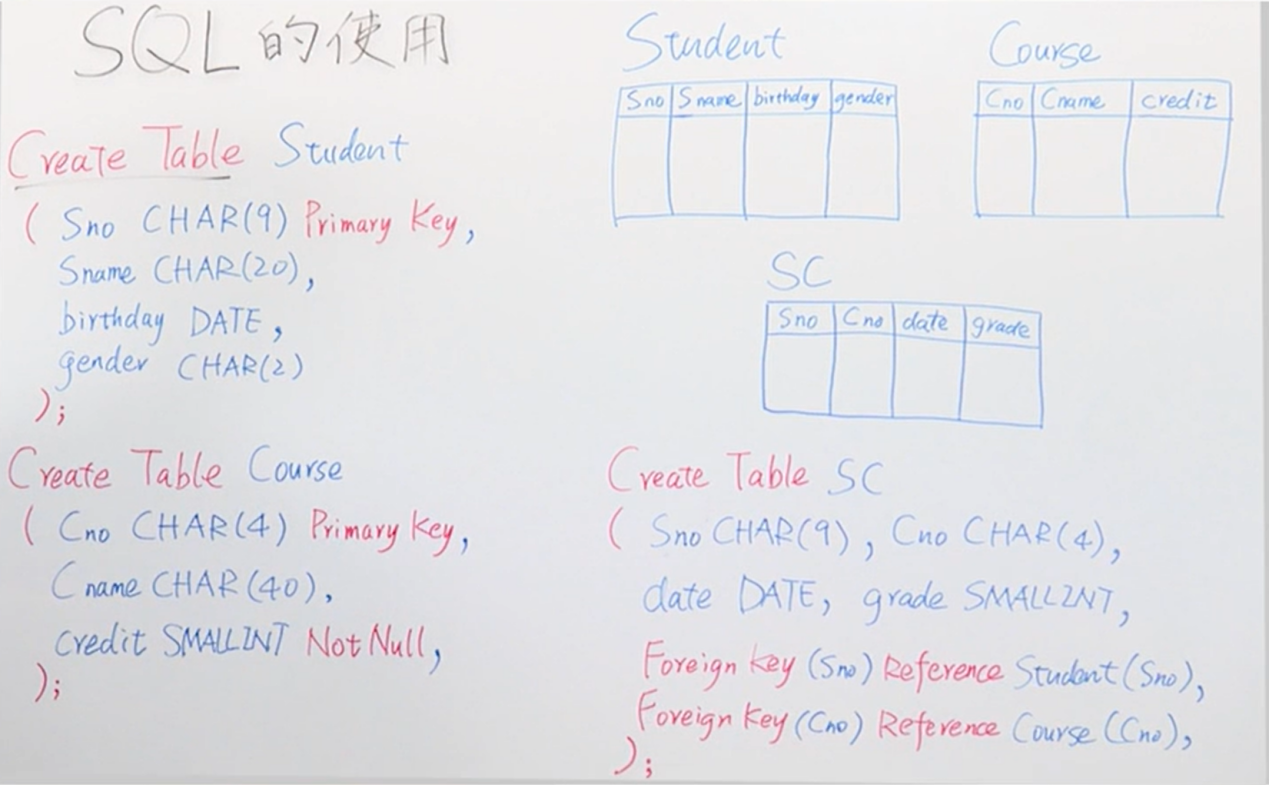
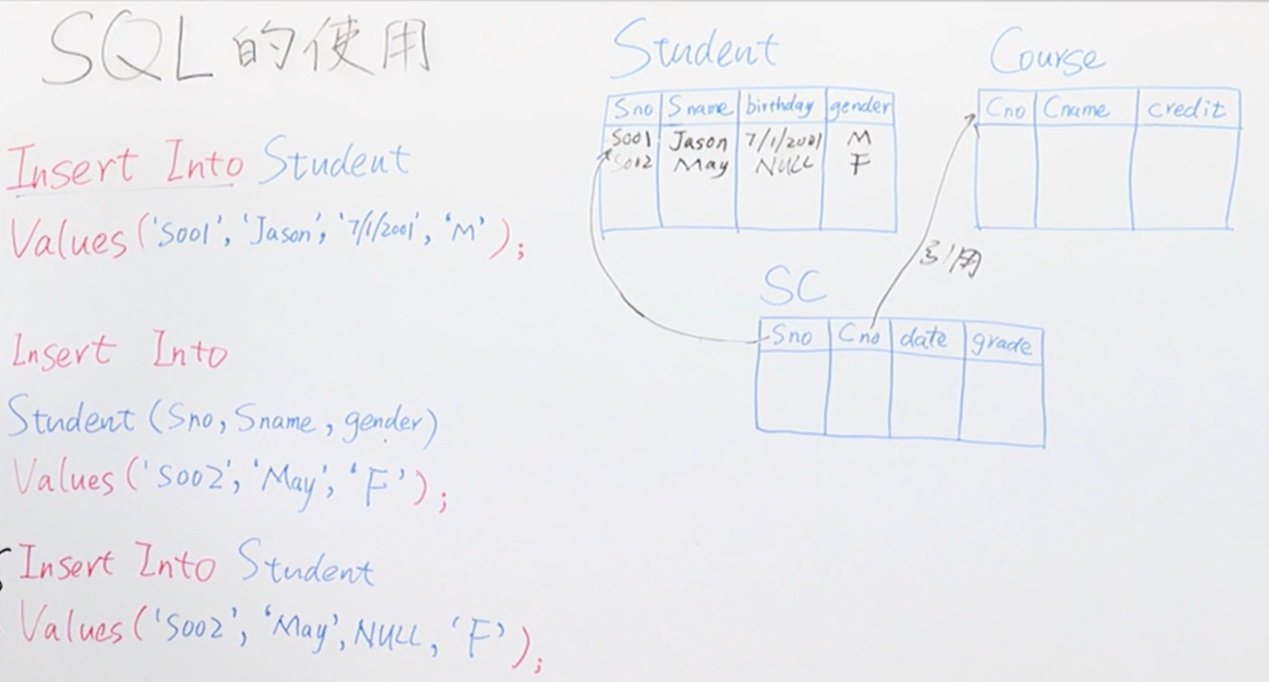
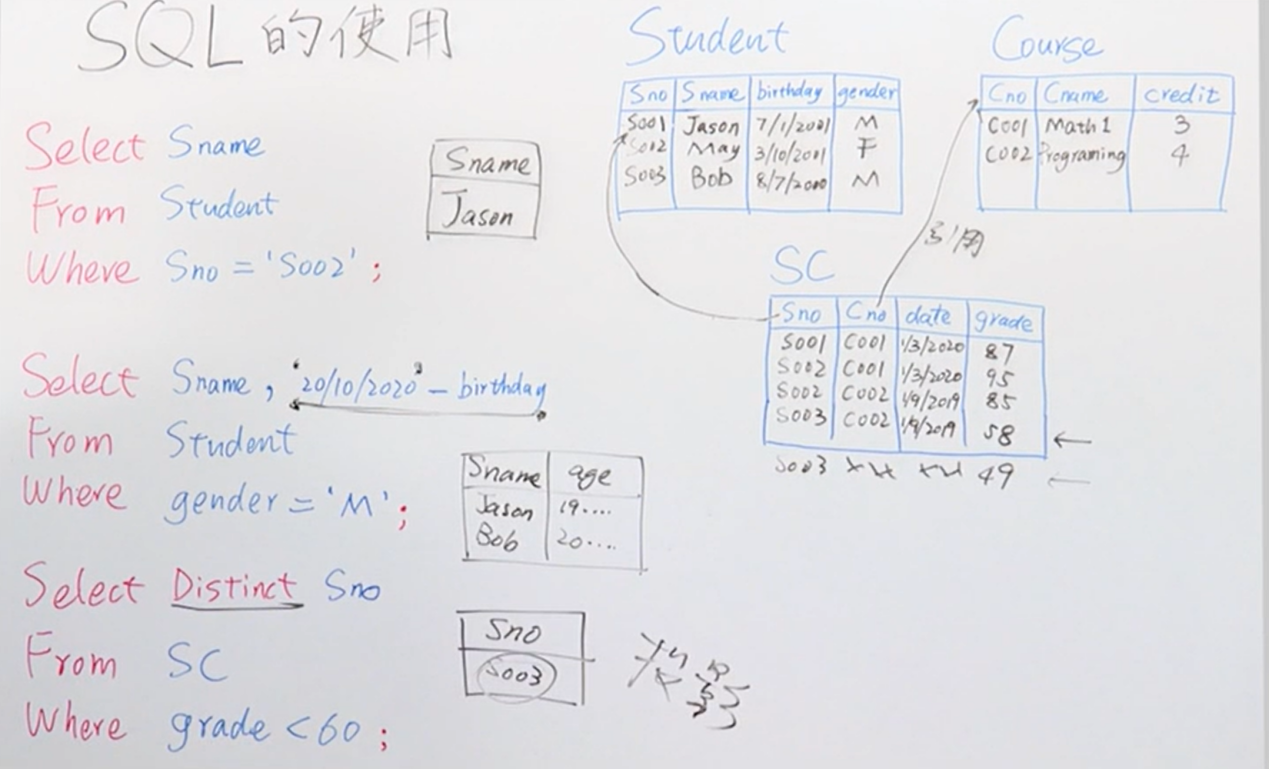
* 关系数据库访问接口 / Interface of RDB
* SQL的使用（CRUD）表示方式有区别
  1. DDL：Create
  2. DQL：Read
  3. DML：Update，Delete
* DDL
  1. 
  2. 约束：
     + Private Key（主码/主键）：唯一识别 只有编号全部不重复，可以作为识别的标准
     + Not Null：数据不为空
     + Foreign Key（外码/外键）：引用 不能引用不在原表中的数据
* 插入数据
  1. 
* SQL单表查询
  1. 

查询出现同样的学号时需要进行去重

**第*1*题：外键约束起哪些作用？（1）告诉数据库系统数据中的引用关系；（2）明确外键所在元组存在依赖于它指向的主键的存在；（3）告诉数据库系统外键指向的主键不能被随意删除。**

A：（1）（2）（3）√

B：（1）（2）

C：（1）（3）

D：（2）（3）

**第*2*题：对于一张学生表Student (s\_no, s\_name, birthday, gender)，我们可以将主键定义在单个属性s\_no上，也可以将定义在复合属性 (s\_no,s\_name) 上。请问哪种方式更优？**

A：s\_no作为主键更优 √

B：(s\_no,s\_name)作为主键更优

C：两者各有优势

D：以上说法均不对

**第*3*题：你觉得在什么样的属性上适合使用Not Null这样的约束？**

A：一个人的住址

B：学生选课表中的成绩

C：一本出版物的出版社 √

D：一门课的教材

**第*4*题：关于SQL查询和关系代数之间的关系，以下哪个说法是不正确的？**

A：一个SQL查询的结果不一定是一个关系

B：求表T在属性a,b上的投影，应使用查询 Select a,b From T √

C：求表T在属性a,b上的投影，应使用查询 Select Distinct a,b From T

D：求条件a>10在表T上的选择结果，应使用查询 Select \* From T Where a>10

作业：

1. 关系数据库让用户自行定义每一张表的Primary Key（主键），用于唯一识别表中的每一行数据。例如学生表student(sno, sname, birthday, gender)的主键可定义为为sno，宿舍表rooms(dorm\_no, room\_no, size, floor)的主键可定义为(dorm\_no, room\_no)（由宿舍号和房间号组成的复合主键）。在文档数据库中，用户无需定义主键，每一个文档都可以由系统自动产生的省缺ID进行识别。换句话说，文档数据库的ID属性起到了Primary Key的作用。请思考：关系数据库的Primary Key机制和文档数据库的ID机制有什么不同，各自的优缺点是什么？

a) 关系数据库的Primary Key机制需要人为指定作为唯一辨认的属性，这个属性一般有含义；文档数据库的ID机制是自动生成的作为唯一辨认的属性，这个属性只有辨认的作用，没有其他含义。

b) Primary Key的优点在于可以从逻辑上选择需要作为主键的有意义的属性，并且可以依据需求设置复合主键，可操控性强，缺点在于需要人为指定，人为选择合适的属性作为Primary Key

c) ID的优点在于其本身的创建和对其索引的创建都由关系数据库系统自动完成，不需要用户手动指定，缺点在于只使用ID机制时，不能从逻辑上对数据做唯一的标记。

2. 关系数据库要求用户在使用一张表之前用DDL对表进行事先定义，并且给出表需满足的各种约束（比如主键、外键等）。文档数据库则不同，它通常不要求用户对文档集的结构做事先定义，甚至允许用户往文档集中插入任意结构的文档。请思考：关系数据库和文档数据库为什么使用了两种不同的功能设计？背后的原因是什么？

a) 因为关系数据库和文档数据库的存储方式，使用方式等都有不同。

b) 关系数据库里的数据按行存储，每一行是一条记录，每一列代表了记录中的某一个属性，这种表型结构需要基于预定好的结构进行数据的操作，因此在使用关系数据库前需要先用DDL对其事先定义。这样可以使数据有良好的结构和规范性。

c) 文档数据库里的数据依文档存储，每一条记录都是一个文档，使用类似对象的方式进行使用，因此用户在插入文档时可以自由选择结构，不需要为文档集的结构做事先定义。这样提高了数据库存储的灵活性。

3. 如果我们在属性A上定义了Primary Key或Unique这样的约束，那么数据库将要求表中的任何一个元组在A上的取值都是唯一的。也就是说，当我们往表中重复插入A值时，数据库将立即感知到，并禁止我们这样做。请问：数据库是如何感知我们往表中重复插入A值的？

a) 可以在属性A上创建索引，使用B树或者哈希表等结构存储所有元组的属性A的内容，插入新数据时，可以将插入的属性A的值快速地和已有的A值作比较，判断是否往表中重复插入了A值。