1. 计算机存储字符（一个数据库只能用一个编码）
2. 如何储存英文字符

ASCII：共有128个，对所有英文字母和字符进行了编码。

Latin-1：共有256个，兼容ASCII码，同时对欧洲符号进行了编码

1. 如何储存中文字符

GB2312：对常用的6千多汉字进行编码，兼容ASCII码

GBK：对2万多汉字进行编码，同时兼容ASCII和GB2312

1. 综合世界上主流国家常用语言进行的编码

Unicode：只兼容ASCII，不兼容GB231、GBK等。具体分为三种存出方案：utf-8,utf-16,utf-32存储方案。

1. 解决MySQL中文乱码

由于MySQL默认编码为Latin-1，解决中文乱码需要更改编码方式。则服务器，客户端与脚本都要更改为UTF8

1. 脚本 .sql 文件另存为时下方选择UTF8编码
2. 更改客户端连接服务器端使用的编码，进入脚本编辑页，在第一行添加

SET NAMES UTF8;

1. 更改服务器端创建数据库使用的编码

CHARSET=UTF8；

举例：

#设置客户端连接服务器端的编码为utf8

SET NAMES UTF8;

DROP DATABASE IF EXISTS xz;

CREATE DATABASE xz CHARSET=UTF8;

USE xz;

CREATE TABLE user(

uid INT,

uname VARCHAR(12),

upwd INT,

email VARCHAR(24),

phone INT,

sex VARCHAR(1),

userName VARCHAR(12),

regTime VARCHAR(8),

isOnline VARCHAR(1)

);

INSERT INTO user VALUES('1','doomfist','5','Talon@qq.com','321123','M','tiequan','20180101','Y');

INSERT INTO user VALUES('2','genji','4','ShimadaClan@163.com','123456','M','yuanshi','20180101','N');

INSERT INTO user VALUES('3','mccree','3','Ash@qq.com','999999','M','makelei','20180101','N');

INSERT INTO user VALUES('4','dva','2','dva@outlook.com','233333','F','DVA','20180101','Y');

INSERT INTO user VALUES('5','widowmaker','1','syr@qq.com','961014','F','wolaopo','19961014','Y');

SELECT \* FROM user;

1. MySQL中的**列类型**（CREATE TABLE xx (nid 列类型)）

TB GB MB KB 1Byte（字节） = 8Bit（位）

1. 数值型——引号可加可不加

提示00开头会自动省略，则只能用字符串

**整型：**

TINYINT 微整型，占1个字节，范围 -128~127（首位为符号位）

SMALLINT 小整型，占2个字节，取值范围 -32768~32767

INT 整型，占4个字节，取值范围 -2147483648~2147483647

BIGINT 大整型，占8个字节，取值范围灰常大

**浮点型：**

FLOAT 单精度浮点型，占4个字节，最多3.4E38，可能会产生计算误差

DOUBLE 双精度浮点型，占8个字节，范围比BIGINT 大得多，可能产生计算误差

DECIMAL（M，D）

定点小数，不会产生计算误差。M代表总有效位数（不包含小数点），D代表小数点后的有效位数

BOOL 布尔型，只有两个值 TRUE、FALSE，真正存储时，列类型会变成TINYINT，T和F会变成1和0，使用时也可直接插入1和0，常用只有两个值的数据。（※T和F使用时不能加引号）

1. 日期型——必须加引号

DATE 日期型 ‘2019-12-31’

TIME 时间型 ‘ 14:53:30’ 有一个空格

DATETIME 日期时间型 ‘2019-12-31 14:53:30’

1. 字符串类型（中文只占1字符，字符不是字节！！！M表示字符数）——必须加引号

VARCHAR（M） 变长字符串，不会产生空间浪费，操作速度较慢，M最大值是65535

CHAR（M） 定长字符串，会产生空间浪费，操作速度较快

M最大值255，一般用于储存固定长度数据

TEXT（M） 大型变长字符串，M最多2G，基本不用

|  |  |  |
| --- | --- | --- |
|  | CHAR(5) | VARCHAR(5) 多的用一个\0代替 |
| a | a\0\0\0\0 | a\0 |
| ab | ab\0\0\0 | ab\0 |
| abcde | abcde | abcde |
| 一二 | 一二\0\0\0 | 一二\0 |

1. MySQL中的**列约束**

MySQL在插入数据时，进行特定的验证；只有满足条件才允许插入，否则被认为是非法插入。 共6种

CREATE TABLE xx( nid 列类型 列约束 )

1. **主键约束**——PRIMARY KEY

声明了主键约束列中，不允许出现重复值，且一个表仅能存在一个主键，通常加在编号列（添加了主键的数字列在查询记录时，会按照编号从小到大排序，加快查找速度，字符串类则不会排序），主键约束列不能加入NULL值。

NULL 表示空，在插入数据时，无法确定要保存的值。NULL和任何值比较都不等，包括两个NULL比较。 ※不能加引号

1. **唯一约束**——UNIQUE