



# 编程三要素 函式、迴圈、判断式

JavaScript 实做  
VISA 与 MASTER CARD 信用卡验证器



# 先备知识

- 对 JavaScript 的语法有基本的认识
  - 宣告变数。
  - 了解基本資料型態： string , numbers, integer, floating-point, array, boolean
  - 加减乘除， 求余数， 比较 ..... 等运算方法。
  - 函式的基本应用， 如何传递参数， 如何定义函式， 如何使用函式， 知道一些基本函式如 console.log() ， alert() ， Number()... 等
  - 清楚迴圈， 判断式的写法。
- 本节课程着重在编程思维与迴圈， 函式， 条件式三者的综合运用。



# 函式

## 1. 代码复用

避免重复代码出现，使程式更简洁

## 2. 功能模组化，结构化

使代码结构清晰，易于阅读、维护，方便团队协作

## 3. 作为程式元件，完成特殊功能。

如实现递归演算法等功能



# 判断式

逻辑判断

依条件作出选择

# 迴圈

在给定范围内进行重複性运算



# 编程思考

- 明确目标
- 定立大纲
- 大问题化小问题，小问题就没问题



# VISA 与 MASTER 信用卡验证

- 如果从左边算起第一个数字是 5 则发卡行为 Master Card，4 则为 Visa，其他数字则判定为不合法。
- 前 15 位数的奇数位经过经过 2 加权后加总（每一个数字都乘以 2 后加总），计算的过程中如果某位数加权后的结果比 10 大，请减去 9。
- 前 15 位数的偶数位加总。
- 奇数位与偶数位算出两个加总的数字相加，将相加后的数字除以 10 取余数，如果结果是 0，那检查码就是 0，否则就是用 10 减去余数的结果。
- 卡号的第 16 码如果与检查码相同，这个卡号就是合法的 VISA 或 MASTER CARD 卡号。



# VISA 与 MASTER 信用卡验证

- 如果从左边算起第一个数字是 5 则发卡行为 Master Card，4 则为 Visa，其他数字则判定为不合法。
- 前 15 位数的奇数位经过经过 2 加权后加总（每一个数字都乘以 2 后加总），计算的过程中如果某位数加权后的结果比 10 大，请减去 9。
- 前 15 位数的偶数位加总。
- 奇数位与偶数位算出两个加总的数字相加，将相加后的数字除以 10 取余数，如果结果是 0，那检查码就是 0，否则就是用 10 减去余数的结果。
- 卡号的第 16 码如果与检查码相同，这个卡号就是合法的 VISA 或 MASTER CARD 卡号。



以下面这个卡号为例

4197124351410926





以下面这个卡号为例

**4**197124351410926

第一位数字为 4 ， 所以是 VISA 卡



以下面这个卡号为例

• 4 1 9 7 1 2 4 3 5 1 4 1 0 9 2 6

将所有奇数位分别乘以 2，若大于 10 则减 9

8, (18 - 9), 2, 8, (10 - 9), 8, 0, 4

将算出来的数字相加：

$$8+9+2+8+1+8+0+4 = 40$$



以下面这个卡号为例

• 4 1 9 7 1 2 4 3 5 1 4 1 0 9 2 6

将所有偶数位加总：

$$1+7+2+3+1+1+9=24$$

将上面两个加总的数字相加：

$$40+24=64$$



以下面这个卡号为例

• 4197124351410926

将刚刚求得的数字除以 10 取余数：

$$64 \% 10 = 4$$

将 10 减去求得的余数：

$$10 - 4 = 6$$



以下面这个卡号为例

• 4197124351410926

6 与卡号尾码相同，所以这个号码是合法的 VISA 卡号

（这里的“合法”卡号只是代表它通过基本演算法验证，并非真实可用于消费之信用卡号。）



# 先写大纲

- 检查卡号是否正确
  - 判断发卡行
  - 计算奇数位加权后加总
    - 判断某位数是否大于 10
  - 计算偶数位加总
  - 计算出检查码
  - 判断检查码是否正确



# 写出执行方法

- 检查卡号是否正确
  - 判断发卡行（if 第一位 = 5 Master, else if 第一位 = 4 VISA, else 不合法）
  - 计算奇数位加权后加总（以 for 迴圈将判断过后的数字加总）
    - 判断某位数是否大于 10（if 某位数 >10 减 9, else 不需处理）
  - 计算偶数位加总（以 for 迴圈将数字加总）
  - 计算出检查码（将加总的两个数字相加后除以 10 取余数，并以 10 减去此余数）
  - 判断检查码是否正确（if 检查码 = 第 16 位数 卡片合法, else 不合法）

# 开始编程



线上即时协作编程工具: [CodeSandbox](#)

信用卡号产生器: [Dummy / Fake Credit Card Generator](#)