

5x{}.tw  
五倍紅寶石

<https://5xruby.tw/>

# 簡介

---



居米  
五倍紅寶石  
Ruby on Rails 工程師

jimmy2822@5xruby.tw

# 經歷

---

- ◆ 數位多媒體設計系遊戲組 – 銘傳大學
- ◆ 遊戲企劃 – 傳奇網路遊戲有限公司
- ◆ 遊戲開發者 – 香港商河洛遊戲有限公司
- ◆ Rails Developer – 五倍紅寶石出礦坑

# O.

程式撰寫環境：<https://repl.it/>

## 課前準備

1.

# 終極密碼

終極密碼

# 1-1.

## 終極密碼

```
# 產生密碼
answer_range = 100
magic_number = rand(1..answer_range)

puts "終極密碼，遊戲開始！請輸入 1~#{answer_range} 以內的數字以破解終極密碼。"

try_count = 0

# 遊戲迴圈、玩家輸入結果檢查
while true
    user_input = gets.to_i
    puts "您輸入的密碼為 #{user_input}"

    if user_input == magic_number
        try_count += 1
        puts "恭喜！終極密碼已成功破解！一共嘗試了#{try_count}次"
        break
    elsif user_input > magic_number
        try_count += 1
        puts "密碼錯誤，小於 #{user_input}"
    else
        try_count += 1
        puts "密碼錯誤，大於 #{user_input}"
    end
end
```

1-2.

玩終極密碼

遊戲體驗

# 1-3.

## 產生密碼

```
# 產生密碼
answer_rang = 100
magic_number = rand(1..answer_range)

puts "終極密碼，遊戲開始！請輸入 1~#{answer_rang} 以內的數字以破解終極密碼。"
```

# 1-4.

## 產生密碼

- ◆ 變數 Variable => answer\_range
- ◆ 方法 Method => rand()
- ◆ 型別 Type => 字串輸出

控制台輸出

# 1-5.

puts()  
控制台輸出

# 1-6.

## 控制台輸出

```
# 方法 puts() 可以印出某些東西至控制台  
puts('Hello World')  
  
# 在 Ruby 的世界裡，方法的小括弧可以省略  
puts '請給我更多的口罩'  
  
# 練習 puts 方法，印出一點東西到控制台吧~
```

puts 練習

# 1-7.

產生密碼

變更密碼產生範圍？

```
answer_range = 100
```

# 1-8.

## 指派變數

```
# 指派變數代表一週可購買口罩數量  
weekly_mask_limit = 2  
  
# 練習：如果一週想買 100 個呢？  
  
puts weekly_mask_limit
```

## 指派變數練習

# 1-9.

產生密碼

嘗試改變密碼產生範圍

```
answer_range = 100
```

# 1-10.

產生密碼

亂數生成方法

```
magic_number = rand(1..answer_range)
```

# 1-11.

## 亂數方法

```
# 呼叫 Ruby 內建函式庫的亂數生成方法 rand()  
  
# 亂數生成 1 到 5 其中的一個整數  
puts rand(1..5)  
  
# 練習：如果想亂數生成 3 到 8 呢？
```

[呼叫亂數方法練習](#)

[解答](#)

中場休息 : )

# 2-1

## 運算子

```
user_input == magic_number  
user_input > magic_number
```

# 2-2

## 運算子

```
# 運算子最終會給你一個結果 true or false  
  
bill_age = 20  
sam_age = 26  
tom_age = 20  
  
# 比較運算子  
  
# bill 年齡大於 sam 嗎?  
puts bill_age > sam_age  
  
# bill 年齡小於 sam 嗎?  
puts bill_age < sam_age  
  
# tom 年齡等於 bill 嗎?  
puts tom_age == bill_age  
  
# 練習題：印出 bill 年齡大於等於 sam 的結果  
  
# 邏輯運算子  
# && (且) || (或)  
  
# bill 年齡小於 sam 且 tom 年齡等於 bill  
puts bill_age < sam_age && tom_age == bill_age  
  
# bill 年齡小於 sam 或 bill 年齡大於 sam  
puts bill_age < sam_age || bill_age > sam_age  
  
# 練習題：印出 tom 年齡大於 sam 或 tom 年齡小於等於 bill 的結果
```

# 2-3

## 流程控制

```
if user_input == magic_number
    try_count += 1
    puts "恭喜！終極密碼已成功破解！一共嘗試了#{try_count}次"
    break
elsif user_input > magic_number
    try_count += 1
    puts "密碼錯誤，小於 #{user_input}"
else
    try_count += 1
    puts "密碼錯誤，大於 #{user_input}"
end
```

# 2-4

## 流程控制

```
# 流程控制

# 如果...就...
# 那如果...就...
# 不然...

# 語法結構
# if 條件成立
#   做事情
# elsif 條件成立
#   做事情
# else
#   做事情
# end
```

```
case_today = 5

# 如果今日確診人數...鄉民就會說...
if case_today > 100
    puts '鄉民：都是海龜惹的禍！'
elsif case_today > 50
    puts '鄉民：民生物資搶起來！'
elsif case_today > 10
    puts '鄉民：誠時鐘乃護國神將也～'
else case_today <= 10
    puts '鄉民：快樂出遊囉！～～～'
end

# 練習題：請更改條件印出 腦波太弱拉！～
wave_power = 150

if wave_power < 100
    puts '腦波太弱拉！～'
else
    puts '就這樣被你增幅！'
end
```

## 流程控制練習

## 解答

# 2-5

迴圈

```
while true
    user_input = gets.to_i
    puts "您輸入的密碼為 #{user_input}"

    if user_input == magic_number
        try_count += 1
        puts "恭喜！終極密碼已成功破解！一共嘗試了#{try_count}次"
        break
    elsif user_input > magic_number
        try_count += 1
        puts "密碼錯誤，小於 #{user_input}"
    else
        try_count += 1
        puts "密碼錯誤，大於 #{user_input}"
    end
end
```

# 2-6

## 迴圈

```
# while 迴圈
# 當條件滿足時，不斷重複執行迴圈

# 當血量大於零時，扣一滴血
hp = 30

while hp > 0
    puts hp -= 1
end

# 使用關鍵字 break 強制跳出迴圈
while true
    puts '我要跳出囉！'
    break
end

# 練習題：如何讓星星符號印出正三角型？
i = 11

while i > 0
    i -= 1
    puts '*' * i
end
```

[迴圈練習](#)      [解答](#)

# 2-7

## 取得輸入

```
# gets 方法可以取得輸入值後回傳字串  
puts "請輸入您的姓名："  
  
# 取得輸入後，將字串指派給變數  
my_name = gets  
  
# 使用字串安插，將變數插入在字串中一起印出  
puts "你好啊，#{my_name}"  
  
# 練習題：使用者輸入數字，比 5 大印出 "大於五！" 小於或等於 5 印出 "小於等於五！"  
# 提示：我們需要對字串使用方法 to_i 轉為數字型別才能比較數字的大小
```

## 取得輸入練習      解答

# 課程回饋

---



[問卷連結](#)

# 學習資源

---

<https://railsbook.tw/>



五倍品質 · 五倍專業



期待與您的合作

Thank you.