【WiredIn Academy】Let's make a rock-paper-scissors game! (The free trial course)

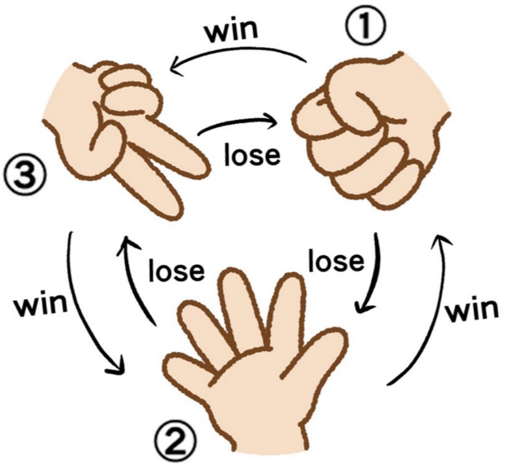
Sample Source URL：https://github.com/techgymjp/techgym_python_en

Execution Environment of program：https://colab.research.google.com/

▪Explanation of rock-paper-scissors game

・What is a rock-paper-scissors game?

A rock-paper-scissors is a game in which the winner is determined by the compatibility of three different hand signs.



The ① hand shape represents a stone.

The ② hand shape represents a paper.

The ③ hand shape represents a scissors.

Stone wins over scissors because stone is hard and cannot be cut with scissors.
Scissors win over paper because scissors can cut paper.
Paper wins over stone because paper can wrap stone.

When both players play the same hand shape, the game is a draw. Then the game is repeated until there is a winner.

The rock-paper-scissors game can be played by any number of players, but this time we will assume two players. The players are you and the computer.

▪1-1 : add ‘paper’ to ‘rock-scissors-paper’ game without ‘paper’: y5YT.py

【Exercise】

Create a ‘rock-scissors-paper’ game with your choice as ‘paper’

【Executed outcome】

```
Start ‘rock-scissors-paper’ game
Input your hand choice

0=rock, 1=scissors, 2=paper2
Win
```

【Hints】

□ ‘My hand choice’ you have input is correctly input in “player” as 0, 1 or 2.
□ ’Computer's hand choice’ is randomly selected in “computer” as 0, 1 or 2
□ 0=rock, 1=scissors, 2=paper

▪1-2 : Devise determination of the outcome: v7Pi.py

【Exercise】

For possible outcome, Win, Loose and Draw,

There are three (3) way of ‘player”
There are three (3) ways of ‘computer’

Devise the outcome and make
If or elif or else three (3) or four (4) ways

【Executed outcome】

```
Start ‘rock-scissors-paper’
Input your hand
0=rock, 1=scissors, 2=paper1
Draw
```

【Hint】

□ Find regularity between “player” and “computer”

■ 1-3: Functionalization: a5Qm.py

【Exercise】

Functionize the program

The functions to make are as below

□ start_message()
  Parameter: None
  Return statement: None
  Content: Display message of ‘start’

□ get_player()
  Parameter: None
  Return statement: Input value
  Content: Display message of urge to input and input

□ get_computer()
  Parameter: None
  Return statement: Computer's hand value
  Content: Acquire computer's hand randomly

□ view_result()
  Parameter: hand_diff
  Return statement: None
  Content: Display win or loose or draw after judging hand_diff

【Executed outcome】

```
Start ‘rock-scissors-paper’
Input your hand chose
0=rock, 1=scissors, 2=paper1
Draw
```

■ 1-4: Display your hand: gP6s.py

【Exercise】

Display your and your computer's hand, rock, scissors and paper

Functions to make are as below

□ get_hand_name()
  Parameter: hand_number
  Return statement: rock or scissors or paper

Content: return rock or scissors or paper corresponding to function 0 or 1 or 2

- view_hand()
  Parameter: player, computer
  Return statement: None
  Content: Display message such as ‘my hand is rock’, ‘computer's hand is paper’ using two parameters

【Executed outcome】

```
Start ‘rock-scissors-paper’
Input your hand
0=rock, 1=scissors, 2=paper1
My hand is scissors
Computer's hand is paper
win
```

【Hints】

- Put a list of rock, scissors or paper in function ‘hands’
- Hands[0] = rock

- 1-5: Use list in input message: dV9E.py

【Exercise】

Using “hands = ‘rock’, ‘scissors’, ‘paper’” automatically form message input(‘0:rock, 1: scissors, 2:paper’)

【Executed outcome】

```
Start ‘rock-scissors-paper’
Input your hand
0:rock, 1:scissors, 2:paper1
My hand is scissors
Computer's hand is paper
win
```

■ 1-6: Use dictionary for result display: L2rT.py

【Exercise】

Use dictionary for result display.
results={‘win’:’you win’, ‘lose’:’you lose’, ‘draw’:’draw try again’}

Functions to make are as below

- get_result()
  Parameter: hand_diff
  Return statement: ‘win’, ‘lose’ or ‘draw’
  Content: return ‘win’, ‘lose’ or ‘draw’ corresponding to parameter “hand_diff”

【Executed outcome】

```
Start ‘rock-scissors-paper’
Input your hand
0:rock, 1:scissors, 2:paper1
My hand is scissors
Computer's hand is paper
you win
```

▪1-7: [Homework] If result is ‘draw’, repeat the game: Jv5e.py

【Exercise】

Repeat again from ‘input(get_player())’ is it is draw, but after displaying ‘draw’s before that

Repeat ‘input → display result’ until it is decided between win or lose

【Executed Outcome】

```
Start ‘rock-scissors-paper’
Input your hand
0:rock, 1:scissors, 2:paper1
My hand is scissors
Computer's hand is scissors
draw try again
Input your hand
0:rock, 1:scissors, 2:paper1
My hand is scissors
Computer's hand is scissors
draw try again
Input your hand
0:rock, 1:scissors, 2:paper2
My hand is paper
Computer's hand is rock
you win
```

【Hints】

- □ You used dictionary above for this
- □ Use function play() in the main part of the game
- □ Execute play() again in recursion when get_result is draw after executing play()

【WiredIn Academy】weekday（9:00–12:00 & 14:00-17:00）Saturday（9:00-12:00）opening a school

・Please ask your trainer for an application!