

遊戲程式設計_精靈寶可夢

一. 寶可夢卡牌角色、設定

【原本玩家擁有的寶可夢】

1. 皮卡丘

屬性：電

HP：3

攻擊：4

防禦：3

速度：6



2. 伊布

屬性：一般

HP：4

攻擊：4

防禦：3

速度：4



3. 胖丁

屬性：妖精

HP：5

攻擊：2

防禦：1

速度：1



4. 基拉祈

屬性：超能力

HP：6

攻擊：6

防禦：6

速度：6



【地圖上待收服寶可夢】

1. 耿鬼

屬性：毒

HP：4

攻擊：4

防禦：4

速度：7

(弱點：超能力-基拉祈)



2. 妙蛙種子

屬性：草

HP：3

攻擊：3

防禦：3

速度：3

(弱點：毒-耿鬼、火-小火龍)



3. 可達鴨

屬性：水

HP：3

攻擊：4

防禦：3

速度：4

(弱點：電-皮卡丘、草-妙蛙種子)



4. 小火龍

屬性：火

HP：3

攻擊：4

防禦：3

速度：4

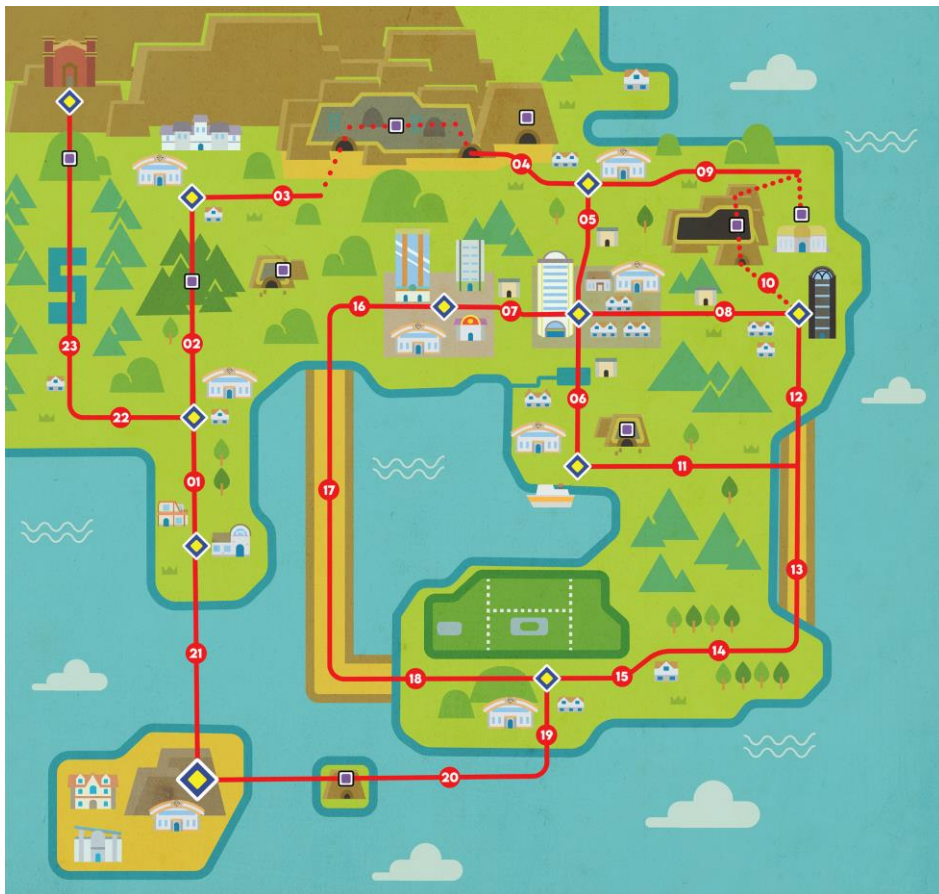
(弱點：水-可達鴨)



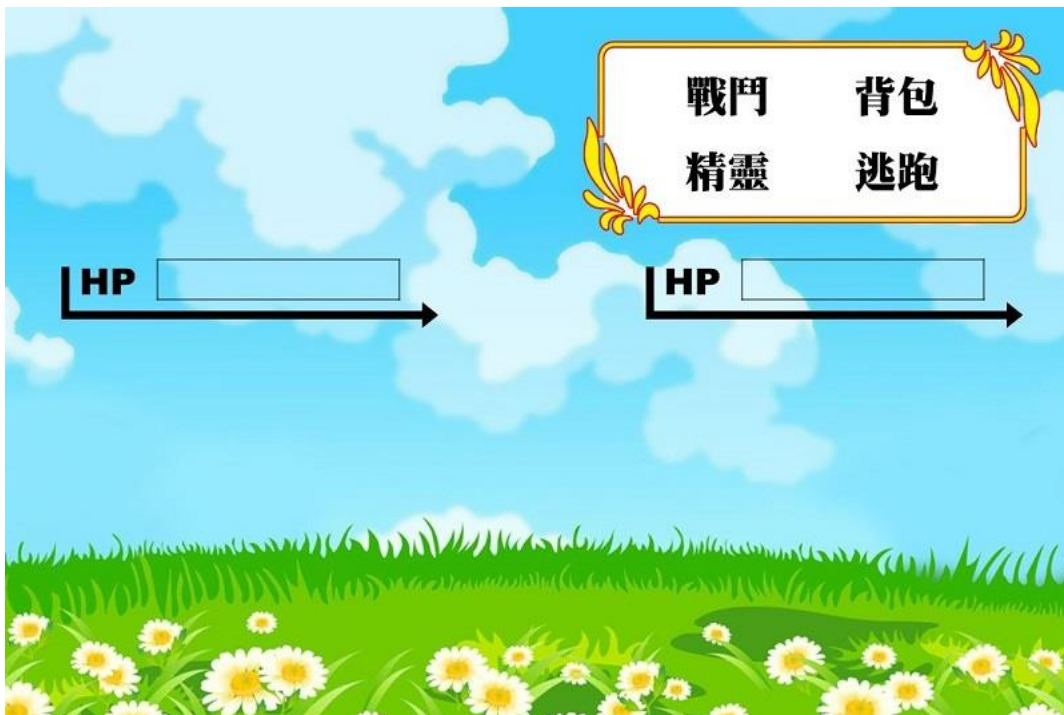
* 註：弱點的依據為官方提供的相剋屬性表

二. 遊戲中用到圖片

- 街道地圖背景



- 戰鬥背景







● 地圖介面相關圖示

地圖背包圖示	背包內的精靈球圖片
	
地圖卡牌圖示	卡包中卡牌圖片
	  <p>以皮卡丘為例 使用該卡片版型，加上上方神奇寶貝圖 示，所製成用在遊戲中的卡牌</p>
地圖探險問號圖示	
	

* 註：其他寶可夢卡牌



● 相關提示圖示

成功收服圖示	未能收服圖示
	
撿到寶貝球圖示	已沒有寶貝球圖示
	

三. 遊戲流程/玩法

- 滑鼠點擊地圖上背包圖示，查看背包內的精靈球
- 滑鼠點擊地圖上卡牌圖示，查看卡包內的精靈卡牌
- 滑鼠點擊問號圖示，可以進入路上遇到精靈模式或撿到寶貝球模式

✧ 路上遇到精靈模式

- ◆ 選擇戰鬥：根據雙方屬性扣除不同血量 → 玩家贏：對話框提示可以收服 → 玩家收服 → 選擇背包，跳出背包視窗，點擊精靈球 → 跳出收服成功提示框、回主畫面、跳出最新卡包視窗
- ◆ 選擇精靈：跳出卡包視窗 → 玩家選擇精靈 → 戰鬥視窗更新完畢
- ◆ 選擇背包：用血量判斷是否收服
 - 能收服：跳出背包視窗，點擊精靈球 → 跳出收服成功提示框、回主畫面、跳出最新卡包視窗
 - 不能收服：跳出未能收服視窗
- ◆ 選擇逃跑：回到主地圖

✧ 路上遇到精靈模式：

未撿到過：跳出撿到寶貝球提示視窗、跳出最新背包狀況、跳出最新地圖狀況

已撿到：跳出已經沒有寶貝球提示視窗

四. 程式碼

- Map.java (地圖視窗，由該檔案執行整個遊戲)

```
1 import java.awt.event.ActionEvent;
2 import java.awt.event.ActionListener;
3
4 import javax.swing.ImageIcon;
5 import javax.swing.JButton;
6 import javax.swing.JFrame;
7 import javax.swing.JLabel;
8 import javax.swing.JPanel;
9 import javax.swing.SwingUtilities;
10
11 public class map extends JFrame implements ActionListener{
12
13     // 設置關卡按鈕
14     JButton[] jbtn = new JButton[7];
15     JButton backpack_jbtn, card_jbtn;
16
17     // 設置圖示路徑
18     String backpack_path = "images/backpack.jpg";
19     ImageIcon bp_icon = new ImageIcon(backpack_path);
20
21     String card_path = "images/card.png";
22     ImageIcon card_icon = new ImageIcon(card_path);
23
24     String question_path = "images/question.jpg";
25     ImageIcon question_icon = new ImageIcon(question_path);
26
27     // 卡牌參數，w耿鬼、x妙蛙種子、y可達鴨、z小火龍
28     public int w, x, y, z;
29     // 寶貝球參數
30     int n, m, l, ballnum;
31
32     public map(int g, int p, int j, int c, int q, int r, int s, int bn) {
33
34         super("精靈寶可夢");
35         setSize(650, 615);
36         setLocation(200, 50);
37
38         // 背景設置
39         String path = "images/map.png";
40         ImageIcon background = new ImageIcon(path);
41         JLabel label = new JLabel(background);
42         label.setBounds(0, 0, this.getWidth(), this.getHeight());
43         JPanel imagePanel = (JPanel) this.getContentPane();
44         imagePanel.setOpaque(false);
45         this.getLayeredPane().add(label, new Integer(Integer.MIN_VALUE));
46         setVisible(true);
47         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
48         imagePanel.setLayout(null);
49
50         // 將傳入的參數進行存取
51         w = g;
52         x = p;
53         y = j;
54         z = c;
55
56         n = q;
57         m = r;
58         l = s;
59         ballnum = bn;
60     }
61 }
```



```

62 // 遇到精靈按鈕
63 jb[0] = new JButton();
64 jb[0].setIcon(question_icon);
65 jb[0].setBounds(115,320,30,30);
66
67
68 jb[1] = new JButton();
69 jb[1].setIcon(question_icon);
70 jb[1].setBounds(115,225,30,30);
71
72 jb[2] = new JButton();
73 jb[2].setIcon(question_icon);
74 jb[2].setBounds(178,117,30,30);
75
76 jb[3] = new JButton();
77 jb[3].setIcon(question_icon);
78 jb[3].setBounds(346,95,30,30);
79
80 jb[4] = new JButton();
81 jb[4].setIcon(question_icon);
82 jb[4].setBounds(485,196,30,30);
83
84 jb[5] = new JButton();
85 jb[5].setIcon(question_icon);
86 jb[5].setBounds(386,259,30,30);
87
88 jb[6] = new JButton();
89 jb[6].setIcon(question_icon);
90 jb[6].setBounds(539,376,30,30);
91
92 // 背包按鈕
93 backpack_jb = new JButton();
94 backpack_jb.setBounds(465, 517, 60, 60);
95 backpack_jb.setIcon(bp_icon);
96 backpack_jb.addActionListener(this);
97 backpack_jb.setVisible(true);
98
99 // 看自己有多少卡牌按鈕
100 card_jb = new JButton();
101 card_jb.setBounds(550, 517, 60, 60);
102 card_jb.setIcon(card_icon);
103 card_jb.addActionListener(this);
104 card_jb.setVisible(true);
105
106 // 按鈕放到地圖Panel上
107 imagePanel.add(backpack_jb);
108 imagePanel.add(card_jb);
109
110 for(int i=0; i<jb.length; i++) {
111     imagePanel.add(jb[i]);
112     jb[i].addActionListener(this);
113     jb[i].setVisible(true);
114 }
115
116 public static void main(String[] args) {
117     map mp = new map(0, 0, 0, 0, 0, 0, 0, 0);
118 }
119
120

```

```

121 @Override
122 public void actionPerformed(ActionEvent evt) {
123     JButton source_jb =(JButton)evt.getSource();
124
125     if(source_jb == backpack_jb) {
126         System.out.println("包包");
127         // 跳出包包查看精靈球
128         new backpack_map(ballnum);
129     }
130
131     else if(source_jb == card_jb) {
132         System.out.println("精靈卡");
133         //跳出卡包查看精靈卡
134         new card(w,x,y,z);
135     }
136
137     // 遇到精靈行戰鬥
138     // 0耿鬼、2妙蛙種子、4可達鴨、6小火龍
139     else if(source_jb == jb[0]) {
140         battle bt = new battle(1,w,x,y,z, n, m ,1, ballnum);
141     }
142
143     else if(source_jb == jb[2]) {
144         battle_2 bt = new battle_2(1,w,x,y,z, n, m ,1, ballnum);
145     }
146
147     else if(source_jb == jb[4]) {
148         battle_4 bt = new battle_4(1,w,x,y,z, n, m ,1, ballnum);
149     }
150

```



```

151     else if(source_jb == jb[6]) {
152         battle_6 bt = new battle_6(1,w,x,y,z, n, m ,1, ballnum);
153     }
154
155     // 檢到寶貝球
156     // n, m, l值進行是否檢到球判斷，0未檢到、1已檢到
157     else if(source_jb == jb[1]) {
158         if(n == 0) {
159             n = 1;
160             new getball(w, x, y ,z, n, m, 1, ballnum);
161         }else if(n == 1) {
162             new notgetball();
163         }
164
165     }
166     else if(source_jb == jb[3]) {
167         if(m == 0) {
168             m = 1;
169             new getball(w, x, y ,z, n, m, 1, ballnum);
170         }else if(m == 1) {
171             new notgetball();
172         }
173     }
174     else if(source_jb == jb[5]) {
175         if(l == 0) {
176             l = 1;
177             new getball(w, x, y ,z, n, m, 1, ballnum);
178         }else if(l == 1) {
179             new notgetball();
180         }
181     }
182 }
183 }

```

- Battle.java (戰鬥視窗)

(另有battle_2, battle_4, battle_6，僅為遇到的精靈不同，因此僅貼上battle程式作為代表)

```
1 import javax.swing.*;
2 import java.awt.*;
3 import java.awt.event.*;
4
5 public class battle extends JFrame implements ActionListener {
6
7     // 設置按鈕，1~12生命值、13戰鬥按鈕、14背包按鈕、15精靈按鈕、16逃跑按鈕
8     public JButton BTN1, BTN2, BTN3, BTN4, BTN5, BTN6, BTN7, BTN8, BTN9, BTN10, BTN11, BTN12, BTN13, BTN14, BTN15, BTN16, cha_1_BTN;
9
10    // 我方使用精靈，預設為皮卡丘，使用路徑為1
11    public String cha_2_path = null;
12    public static int path = 1;
13
14    // 卡牌參數，w耿鬼、x妙蛙種子、y可達鴨、z小火龍
15    public int a = 1, w = 0, x = 0, y = 0, z = 0;
16    // 寶貝球參數
17    int n, m, l, ballnum;
18
19    JPanel imagePanel;
20
21    public battle(int path, int g, int p, int j, int c, int q, int r, int s, int bn) {
22
23        super("戰鬥");
24        setSize(800, 540);
25        setLocation(200, 50);
26
27        // 背景設置
28        String bg_path = "images\\battle_background.jpg";
29        ImageIcon background = new ImageIcon(bg_path);
30        JLabel label = new JLabel(background);
31        label.setBounds(0, 0, this.getWidth(), this.getHeight());
32        imagePanel = (JPanel) this.getContentPane();
33        imagePanel.setOpaque(false);
34        this.getLayeredPane().add(label, new Integer(Integer.MIN_VALUE));
35        setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
36
37        // 呼叫按鈕函式，放置按鈕
38        button(path);
39        setVisible(true);
40
41        // 將傳入的參數進行存取
42        w = g;
43        x = p;
44        y = j;
45        z = c;
46
47        n = q;
48        m = r;
49        l = s;
50        ballnum = bn;
51    }
```

```

53 // 設置按鈕排版
54 public void button(int path){
55     imagePanel.setLayout(null);
56
57     BTN1 = new JButton("");
58     BTN1.setSize(30, 30);
59     BTN1.setLocation(115,193);
60     BTN1.setBackground(Color.GREEN);
61     imagePanel.add(BTN1);
62     BTN1.setVisible(true);
63
64     BTN2 = new JButton("");
65     BTN2.setSize(30, 30);
66     BTN2.setLocation(115 + 30,193);
67     BTN2.setBackground(Color.GREEN);
68     imagePanel.add(BTN2);
69     BTN2.setVisible(true);
70
71     BTN3 = new JButton("");
72     BTN3.setSize(30, 30);
73     BTN3.setLocation(115 + 60,193);
74     BTN3.setBackground(Color.GREEN);
75     imagePanel.add(BTN3);
76     BTN3.setVisible(true);
77
78     BTN4 = new JButton("");
79     BTN4.setSize(30, 30);
80     BTN4.setLocation(115 + 90,193);
81     BTN4.setBackground(Color.GREEN);
82     imagePanel.add(BTN4);
83     BTN4.setVisible(true);
84
85     BTN5 = new JButton("");
86     BTN5.setSize(30, 30);
87     BTN5.setLocation(115 + 120,193);
88     BTN5.setBackground(Color.GREEN);
89     imagePanel.add(BTN5);
90     BTN5.setVisible(true);
91
92     BTN6 = new JButton("");
93     BTN6.setSize(30, 30);
94     BTN6.setLocation(115 + 150,193);
95     BTN6.setBackground(Color.GREEN);
96     imagePanel.add(BTN6);
97     BTN6.setVisible(true);
98
99     BTN7 = new JButton("");
100    BTN7.setSize(30, 30);
101    BTN7.setLocation(553,193);
102    BTN7.setBackground(Color.GREEN);
103    imagePanel.add(BTN7);
104    BTN7.setVisible(true);
105
106    BTN8 = new JButton("");
107    BTN8.setSize(30, 30);
108    BTN8.setLocation(553 + 30,193);
109    BTN8.setBackground(Color.GREEN);
110    imagePanel.add(BTN8);
111    BTN8.setVisible(true);
112
113
114    BTN9 = new JButton("");
115    BTN9.setSize(30, 30);
116    BTN9.setLocation(553 + 60,193);
117    BTN9.setBackground(Color.GREEN);
118    imagePanel.add(BTN9);
119    BTN9.setVisible(true);
120
121    BTN10 = new JButton("");
122    BTN10.setSize(30, 30);
123    BTN10.setLocation(553 + 90,193);
124    BTN10.setBackground(Color.GREEN);
125    imagePanel.add(BTN10);
126    BTN10.setVisible(true);
127
128    BTN11 = new JButton("");
129    BTN11.setSize(30, 30);
130    BTN11.setLocation(553 + 120,193);
131    BTN11.setBackground(Color.GREEN);
132    imagePanel.add(BTN11);
133    BTN11.setVisible(true);
134
135    BTN12 = new JButton("");
136    BTN12.setSize(30, 30);
137    BTN12.setLocation(553 + 150,193);
138    BTN12.setBackground(Color.GREEN);
139    imagePanel.add(BTN12);
140    BTN12.setVisible(true);
141
142    BTN13 = new JButton("戰鬥");
143    BTN13.setSize(85, 60);
144    BTN13.setLocation(500, 37);
145    BTN13.setBackground(Color.WHITE);
146    BTN13.setFont(new java.awt.Font("Microsoft JhengHei", 1, 24));
147    BTN13.addActionListener(this);
148    imagePanel.add(BTN13);
149    BTN13.setVisible(true);
150
151    BTN14 = new JButton("背包");
152    BTN14.setSize(85, 60);
153    BTN14.setLocation(620, 37);
154    BTN14.setBackground(Color.WHITE);
155    BTN14.setFont(new java.awt.Font("Microsoft JhengHei", 1, 24));
156    BTN14.addActionListener(this);
157    imagePanel.add(BTN14);
158    BTN14.setVisible(true);
159
160    BTN15 = new JButton("精靈");
161    BTN15.setSize(85, 60);
162    BTN15.setLocation(500, 99);
163    BTN15.setBackground(Color.WHITE);
164    BTN15.setFont(new java.awt.Font("Microsoft JhengHei", 1, 24));
165    BTN15.addActionListener(this);
166    imagePanel.add(BTN15);
167    BTN15.setVisible(true);
168
169    BTN16 = new JButton("逃跑");
170    BTN16.setSize(85, 60);
171    BTN16.setLocation(620, 99);
172    BTN16.setBackground(Color.WHITE);
173    BTN16.setFont(new java.awt.Font("Microsoft JhengHei", 1, 24));
174    BTN16.addActionListener(this);
175    imagePanel.add(BTN16);
176    BTN16.setVisible(true);
177
178    String cha_1_path = "images\\250\\5.png";
179    ImageIcon cha_1 = new ImageIcon(cha_1_path);
180    cha_1_BTN = new JButton(cha_1);
181    cha_1_BTN.setSize(220, 220);
182    cha_1_BTN.setLocation(75,260);
183    cha_1_BTN.addActionListener(this);
184    imagePanel.add(cha_1_BTN);
185    cha_1_BTN.setVisible(true);
186
187    // 傳入路徑判斷，以進行我方寶可夢更換
188    if(path == 1) {
189        cha_2_path = "images\\250\\1.png";
190    }else if(path == 2) {
191        cha_2_path = "images\\250\\2.png";
192    }else if(path == 3) {
193        cha_2_path = "images\\250\\3.png";
194    }else if(path == 4) {
195        cha_2_path = "images\\250\\4.png";
196    }else if(path == 5) {
197        cha_2_path = "images\\250\\5.png";
198    }else if(path == 6) {
199        cha_2_path = "images\\250\\6.png";
200    }else if(path == 7) {
201        cha_2_path = "images\\250\\7.png";
202    }else if(path == 8) {
203        cha_2_path = "images\\250\\8.png";
204    }
205    ImageIcon cha_2 = new ImageIcon(cha_2_path);
206    JButton cha_2_BTN = new JButton(cha_2);
207    cha_2_BTN.setSize(220, 220);
208    cha_2_BTN.setLocation(515,260);
209    imagePanel.add(cha_2_BTN);
210    cha_2_BTN.setVisible(true);
211
212
213
214 @Override
215 public void actionPerformed(ActionEvent e) {
216
217     // 攻擊後的生命值結果設置
218     if(e.getSource() == BTN13) {
219         if(cha_2_path == "images\\250\\1.png") {
220             BTN12.setBackground(Color.WHITE);
221             BTN11.setBackground(Color.WHITE);
222             BTN10.setBackground(Color.WHITE);
223             BTN6.setBackground(Color.WHITE);
224             BTN5.setBackground(Color.WHITE);
225         }else if(cha_2_path == "images\\250\\2.png"){
226             BTN12.setBackground(Color.WHITE);
227             BTN11.setBackground(Color.WHITE);
228             BTN6.setBackground(Color.WHITE);
229             BTN5.setBackground(Color.WHITE);
230         }else if(cha_2_path == "images\\250\\3.png"){
231             BTN12.setBackground(Color.WHITE);
232             BTN11.setBackground(Color.WHITE);
233             BTN10.setBackground(Color.WHITE);
234             BTN9.setBackground(Color.WHITE);
235             BTN6.setBackground(Color.WHITE);
236         }else if(cha_2_path == "images\\250\\4.png"){
237             BTN12.setBackground(Color.WHITE);
238             BTN6.setBackground(Color.WHITE);
239             BTN5.setBackground(Color.WHITE);
240             BTN4.setBackground(Color.WHITE);
241             BTN3.setBackground(Color.WHITE);
242             // 已達可收服標準，跳出可收服提示視窗，將收服的判斷值從1改為0
243             hint.cc = new hint();
244             a = 0;
245         }else if(cha_2_path == "images\\250\\5.png"){
246             BTN12.setBackground(Color.WHITE);
247             BTN11.setBackground(Color.WHITE);
248             BTN6.setBackground(Color.WHITE);
249             BTN5.setBackground(Color.WHITE);
250         }else if(cha_2_path == "images\\250\\6.png"){
251             BTN12.setBackground(Color.WHITE);
252             BTN11.setBackground(Color.WHITE);
253             BTN6.setBackground(Color.WHITE);
254         }else if(cha_2_path == "images\\250\\7.png"){
255             BTN12.setBackground(Color.WHITE);
256             BTN6.setBackground(Color.WHITE);
257         }else if(cha_2_path == "images\\250\\8.png"){
258             BTN12.setBackground(Color.WHITE);
259             BTN6.setBackground(Color.WHITE);
260         }
261     }
262

```

```
262 // 開啟背包，以進行收服
263 else if(e.getSource() == BTN14) {
264     backpack cc = new backpack(a, 4, w, x, y, z, n, m, 1, ballnum);
265 }
266 // 開啟卡牌，更換精靈
267 else if(e.getSource() == BTN15) {
268     this.setVisible(false);
269     pokemon cc = new pokemon(w,x,y,z, n, m, 1, ballnum);
270 }
271 // 逃跑
272 else if(e.getSource() == BTN16) {
273     this.setVisible(false);
274 }
275 // 點擊耿鬼觀看角色訊息
276 else if(e.getSource() == cha_1_BTN) {
277     gengar cc = new gengar();
278 }
279 }
280 }
```

- Pokemon.java (戰鬥更換卡牌視窗)

(另有pokemon_2, pokemon_4, pokemon_6，僅為配合的戰鬥視窗不同，因此僅貼上pokemon作為代表)

```
1 import java.awt.BorderLayout;
2 import java.awt.Color;
3 import java.awt.Dimension;
4 import java.awt.GridLayout;
5 import java.awt.event.ActionEvent;
6 import java.awt.event.ActionListener;
7 import javax.swing.ImageIcon;
8 import javax.swing.JButton;
9 import javax.swing.JFrame;
10 import javax.swing.JLabel;
11 import javax.swing.JPanel;
12 import javax.swing.SwingConstants;
13
14 class pokemon extends JFrame implements ActionListener{
15
16     // 設置卡牌
17     JButton jb[] = new JButton[9];
18
19     // 卡牌參數，w耿鬼、x妙蛙種子、y可達鴨、z小火龍
20     public int w, x, y, z;
21
22     // 寶貝球參數
23     int n, m, l, ballnum;
24
25     public pokemon(int g,int p,int j,int c, int q, int r, int s, int bn) {
26
27         // 設置版型
28         super("精靈");
29         this.setSize(426, 503);
30         this.setLocation(424, 50);
31         setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
32         setVisible(true);
33
34         JPanel jp = new JPanel();
35         GridLayout gl = new GridLayout(3,3);
36         jp.setLayout(gl);
37
38         for(int i=0; i<9; i++) {
39             jb[i] = new JButton();
40             jb[i].setSize(122, 151);
41             jb[i].setBackground(Color.WHITE);
42             jp.add(jb[i]);
43         }
44
45         // 皮卡丘卡牌
46         String card_pikachu_path = "images/card_pikachu.png";
47         ImageIcon card_pikachu_icon = new ImageIcon(card_pikachu_path);
48         jb[0].setIcon(card_pikachu_icon);
49         jb[0].addActionListener(this);
50
51         // 伊布卡牌
52         String card_Eevee_path = "images/card_Eevee.png";
53         ImageIcon card_Eevee_icon = new ImageIcon(card_Eevee_path);
54         jb[1].setIcon(card_Eevee_icon);
55         jb[1].addActionListener(this);
```

```

57 // 胖丁卡牌
58 String card_Jigglypuff_path = "images/card_Jigglypuff.png";
59 ImageIcon card_Jigglypuff_icon = new ImageIcon(card_Jigglypuff_path);
60 jb[2].setIcon(card_Jigglypuff_icon);
61 jb[2].addActionListener(this);
62
63 // 基拉祈卡牌
64 String card_Jirachi_path = "images/card_Jirachi.png";
65 ImageIcon card_Jirachi_icon = new ImageIcon(card_Jirachi_path);
66 jb[3].setIcon(card_Jirachi_icon);
67 jb[3].addActionListener(this);
68
69 // 將傳入的參數進行存取
70 w = g;
71 x = p;
72 y = j;
73 z = c;
74
75 n = q;
76 m = r;
77 l = s;
78 ballnum = bn;
79
80 // 如果收服到耿鬼
81 if(w == 1) {
82     String card_Gengar_path = "images/card_Gengar.png";
83     ImageIcon card_Gengar_icon = new ImageIcon(card_Gengar_path);
84     jb[4].setIcon(card_Gengar_icon);
85     jb[4].addActionListener(this);
86 }
87
88 // 如果收服到妙挖種子
89 if(x == 1) {
90     String card_Bulbasaur_path = "images/card_Bulbasaur.png";
91     ImageIcon card_Bulbasaur_icon = new ImageIcon(card_Bulbasaur_path);
92     jb[5].setIcon(card_Bulbasaur_icon);
93     jb[5].addActionListener(this);
94 }
95
96 // 如果收服到可達鴨
97 if(y == 1) {
98     String card_Psyduck_path = "images/card_Psyduck.png";
99     ImageIcon card_Psyduck_icon = new ImageIcon(card_Psyduck_path);
100     jb[6].setIcon(card_Psyduck_icon);
101     jb[6].addActionListener(this);
102 }
103
104 // 如果收服到小火龍
105 if(z == 1) {
106     String card_Charmander_path = "images/card_Charmander.png";
107     ImageIcon card_Charmander_icon = new ImageIcon(card_Charmander_path);
108     jb[7].setIcon(card_Charmander_icon);
109     jb[7].addActionListener(this);
110 }

```

```

112 // 放上標題
113 JPanel nev_jp = new JPanel();
114 JLabel jl = new JLabel("寶可夢卡牌", SwingConstants.CENTER);
115 jl.setFont(new java.awt.Font("Microsoft JhengHei", 1, 25));
116 nev_jp.add(jl);
117
118 // 畫片排版
119 JPanel lf_jp = new JPanel();
120 JPanel ri_jp = new JPanel();
121
122 lf_jp.setPreferredSize(new Dimension(30, 500));
123 ri_jp.setPreferredSize(new Dimension(30, 500));
124
125 this.add(lf_jp, BorderLayout.WEST);
126 this.add(ri_jp, BorderLayout.EAST);
127 this.add(nev_jp, BorderLayout.NORTH);
128 this.add(jp, BorderLayout.CENTER);
129
130 }
131
132 @Override
133 public void actionPerformed(ActionEvent e) {
134     // 點擊卡牌進行我方角色更換
135     if(e.getSource() == jb[0]) {
136         update(1, w, x, y, z, n, m, l, ballnum);
137     } else if(e.getSource() == jb[1]) {
138         update(2, w, x, y, z, n, m, l, ballnum);
139     } else if(e.getSource() == jb[2]) {
140         update(3, w, x, y, z, n, m, l, ballnum);
141     } else if(e.getSource() == jb[3]) {
142         update(4, w, x, y, z, n, m, l, ballnum);
143     } else if(e.getSource() == jb[4]) {
144         update(5, w, x, y, z, n, m, l, ballnum);
145     } else if(e.getSource() == jb[5]) {
146         update(6, w, x, y, z, n, m, l, ballnum);
147     } else if(e.getSource() == jb[6]) {
148         update(7, w, x, y, z, n, m, l, ballnum);
149     } else if(e.getSource() == jb[7]) {
150         update(8, w, x, y, z, n, m, l, ballnum);
151     }
152 }
153
154 // 更換卡牌函式：卡牌視窗消失，更新戰鬥視窗
155 public void update(int a, int g, int p, int j, int c, int q, int r, int s, int bn) {
156     this.setVisible(false);
157     battle program = new battle(a, g, p, j, c, n, m, l, bn);
158     program.setVisible(true);
159 }
160 }

```


● Backpack.java (戰鬥中的背包視窗・點擊寶貝球以收服神奇寶貝)

```
1 import java.awt.BorderLayout;
2 import java.awt.Color;
3 import java.awt.Dimension;
4 import java.awt.GridLayout;
5 import java.awt.event.ActionEvent;
6 import java.awt.event.ActionListener;
7
8 import javax.swing.ImageIcon;
9 import javax.swing.JButton;
10 import javax.swing.JFrame;
11 import javax.swing.JLabel;
12 import javax.swing.JPanel;
13 import javax.swing.SwingConstants;
14
15 public class backpack extends JFrame implements ActionListener{
16
17     // 設置寶貝球按鈕
18     JButton jb[] = new JButton[9];
19
20     // 設置欲收服的卡牌參數
21     int card_number;
22
23     // 卡牌參數，w耿鬼、x妙蛙種子、y可達鴨、z小火龍
24     public int w, x, y, z;
25
26     // 寶貝球參數
27     int n, m, l, ballnum;
28
29     public backpack(int a, int n_1,int g,int p,int j,int c, int q, int r, int s, int bn) {
30
31         // 版面設置
32         this.setSize(350, 350);
33         this.setLocation(650, 50);
34         this.setVisible(true);
35         this.setTitle("背包");
36
37         // 將傳入的參數進行存取
38         w = g;
39         x = p;
40         y = j;
41         z = c;
42
43         n = q;
44         m = r;
45         l = s;
46         ballnum = bn;
47
48         // 設置寶貝球
49         JPanel jp = new JPanel();
50         jp.setBackground(Color.WHITE);
51         GridLayout gl = new GridLayout(3,3);
52         jp.setLayout(gl);
53
54         String ball_path = "images/ball.jpg";
55         ImageIcon ball_icon = new ImageIcon(ball_path);
```

```

57     for(int i=0; i<jb.length; i++) {
58         jb[i] = new JButton();
59         jb[i].setSize(100, 100);
60         jb[i].setBackground(Color.WHITE);
61         jp.add(jb[i]);
62         // 透過傳入的總寶貝球bn數進行視窗呈現的顆數
63         if(i < (3+bn)) {
64             jb[i].setIcon(ball_icon);
65         }
66     }
67
68     // 收服判斷值為0，可收服
69     if(a == 0) {
70         for(int i=0; i<jb.length; i++) {
71             jb[i].addActionListener(this);
72         }
73     }
74     // 收服判斷值為1，跳出無法收服提示視窗
75     else {
76         notget cc = new notget();
77         this.toFront();
78         this.setVisible(false);
79     }
80
81     // 設置標題
82     JPanel nev_jp = new JPanel();
83     nev_jp.setSize(650, 80);
84     JLabel jl = new JLabel("精靈球", SwingConstants.CENTER);
85     jl.setFont(new java.awt.Font("Microsoft JhengHei", 1, 25));
86     nev_jp.add(jl);
87
88     // 畫片排版
89     JPanel lf_jp = new JPanel();
90     JPanel ri_jp = new JPanel();
91
92     lf_jp.setPreferredSize(new Dimension(25, 350));
93     ri_jp.setPreferredSize(new Dimension(25, 350));
94
95     this.add(lf_jp, BorderLayout.WEST);
96     this.add(ri_jp, BorderLayout.EAST);
97     this.add(nev_jp, BorderLayout.NORTH);
98     this.add(jp, BorderLayout.CENTER);
99
100    // 傳輸值收服卡片的值
101    card_number = n_1;
102 }
103
104 @Override
105 public void actionPerformed(ActionEvent arg0) {
106     // 傳值呼叫get，開啟收服成功提示視窗，與卡牌最新結果視窗
107     get cc = new get(card_number, w, x, y, z, n, m, l, ballnum);
108     this.setVisible(false);
109 }
110
111 }

```

- Hint.java (可進行收服提示視窗)

```
1 import javax.swing.*;
2 import java.awt.*;
3 import java.awt.event.*;
4
5 // 可收服神奇寶貝提示視窗
6 class Hint extends JFrame{
7
8     public Hint() {
9
10         // 版面設置
11         super("提示!");
12         setSize(250,110);
13         setLocation(200, 50);
14         setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
15         setVisible(true);
16
17         JPanel jp = new JPanel();
18         jp.setBackground(Color.WHITE);
19
20         jp.setLayout(new FlowLayout(FlowLayout.CENTER));
21         JLabel label=new JLabel("<html><body><p align=\"center\">對方HP值已低於3<br/>可使用精靈球收服！</p></body></html>");
22         label.setSize(200,200);
23         label.setFont(new java.awt.Font("Microsoft JhengHei", 1, 20));
24         label.setLocation(0, 0);
25         jp.add(label);
26
27         this.add(jp);
28     }
29 }
```

- Get.java (收成功提示窗)

```

1 import javax.swing.*;
2 import java.awt.*;
3 import java.awt.event.*;
4
5 class get extends JFrame{
6
7     // 收服圖示
8     JButton pic;
9
10    // 卡牌參數，w耿鬼、x妙蛙種子、y可達鴨、z小火龍
11    int w,x,y,z;
12
13    // 收服卡片參數n_1，寶貝球參數
14    int n_1, n, m, l, ballnum;
15
16    public get(int n_1,int g,int p,int j,int c,int q, int r, int s, int bn) {
17
18        // 版面設置
19        super("GET!");
20        setSize(400, 400);
21        setLocation(200, 50);
22        setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
23        setVisible(true);
24
25        JPanel jp = new JPanel();
26        jp.setBackground(Color.WHITE);
27
28        jp.setLayout(new FlowLayout(FlowLayout.CENTER));
29        JLabel label = new JLabel("<html><body><p align='center'>您收服了一隻寶可夢！</p></body></html>");
30        label.setSize(200, 200);
31        label.setFont(new java.awt.Font("Microsoft JhengHei", 1, 25));
32        jp.add(label);
33
34        ImageIcon a = new ImageIcon("images\\luffy.gif");
35        pic = new JButton(a);
36        pic.setSize(340, 300);
37        pic.setBackground(Color.WHITE);
38        pic.setVisible(true);
39        jp.add(pic);
40
41        this.add(jp);
42
43        // 判斷收服卡片的值，將傳入的參數進行存取
44        // 收服w耿鬼
45        if(n_1 == 4) {
46            w = 1;
47            x = p;
48            y = j;
49            z = c;
50        }
51
52        // 收服x妙蛙種子
53        if(n_1 == 5) {
54            w = g;
55            x = 1;
56            y = j;
57            z = c;
58        }
59
60        // 收服y可達鴨
61        if(n_1 == 6) {
62            w = g;
63            x = p;
64            y = 1;
65            z = c;
66        }
67
68        // 收服z小火龍
69        if(n_1 == 7) {
70            w = g;
71            x = p;
72            y = j;
73            z = 1;
74        }
75
76        // 將傳入的寶貝球參數進行存取
77        n = q;
78        m = r;
79        l = s;
80        ballnum = bn -1;
81
82        // 更新地圖與卡牌
83        map cc = new map(w,x,y,z,n, m, l, ballnum);
84        card dd = new card(w,x,y,z);
85        // 將成功收服提示視窗擺在最前方
86        this.toFront();
87    }
88 }

```

- Notget.java (未能收服提示視窗)

```
1 import javax.swing.*;
2 import java.awt.*;
3 import java.awt.event.*;
4
5 class notget extends JFrame{
6
7     // 未收服圖示
8     JButton pic;
9
10    public notget() {
11
12        // 版面設置
13        super("警告!");
14        setSize(400, 382);
15        setLocation(200, 50);
16        setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
17        setVisible(true);
18
19        JPanel jp = new JPanel();
20        jp.setBackground(Color.WHITE);
21
22        jp.setLayout(new FlowLayout(FlowLayout.CENTER));
23        JLabel label = new JLabel("<html><body><p align=\"center\">您目前無法收服這隻寶可夢！</p></body></html>");
24        label.setSize(200, 200);
25        label.setFont(new java.awt.Font("Microsoft JhengHei", 1, 25));
26        jp.add(label);
27
28        ImageIcon a = new ImageIcon("images\\chopper.gif");
29        pic = new JButton(a);
30        pic.setSize(340, 300);
31        pic.setBackground(Color.WHITE);
32        pic.setVisible(true);
33        jp.add(pic);
34
35        this.add(jp);
36    }
37 }
```

- Getball.java (撿到寶貝球提示視窗)

```
1 import java.awt.Color;
2 import java.awt.FlowLayout;
3
4 import javax.swing.ImageIcon;
5 import javax.swing.JButton;
6 import javax.swing.JFrame;
7 import javax.swing.JLabel;
8 import javax.swing.JPanel;
9
10 public class getball extends JFrame{
11
12     // 獲得精靈球圖示
13     JButton pic;
14
15     // 卡牌參數，w耿鬼、x妙蛙種子、y可達鴨、z小火龍
16     int w, x, y, z;
17
18     // 寶貝球參數
19     int n, m, l, bn;
20
21     public getball(int g,int p,int j,int c, int q, int r, int s, int bn) {
22
23         // 版面設置
24         super("GET BALL!");
25         setSize(350, 350);
26         setLocation(200, 50);
27         setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
28         setVisible(true);
29
30         JPanel jp = new JPanel();
31         jp.setBackground(Color.WHITE);
32
33         jp.setLayout(new FlowLayout(FlowLayout.CENTER));
34         JLabel label = new JLabel("<html><body><p align=\"center\">您獲得了一顆精靈球！</p></body></html>");
35         label.setSize(200, 200);
36         label.setFont(new java.awt.Font("Microsoft JhengHei", 1, 25));
37         jp.add(label);
38
39         ImageIcon get_ball = new ImageIcon("images\\pokeball.gif");
40         pic = new JButton(get_ball);
41         pic.setSize(340, 300);
42         pic.setBackground(Color.WHITE);
43         pic.setVisible(true);
44         jp.add(pic);
45
46         this.add(jp);
47
48         // 將傳入的參數進行存取
49         w = g;
50         x = p;
51         y = j;
52         z = c;
53
54         n = q;
55         m = r;
56         l = s;
57
58         // 得到寶貝球數+1
59         bn++;
60
61         // 更新地圖與背包
62         new map(w,x,y,z, q, r, s, bn);
63         new backpack_map(bn);
64
65         // 將成功取得提示視窗擺在最前方
66         this.toFront();
67     }
68
69 }
```

- Notgetball.java (已經沒有寶貝球提示視窗)

```
1 import java.awt.Color;
2 import java.awt.FlowLayout;
3
4 import javax.swing.ImageIcon;
5 import javax.swing.JButton;
6 import javax.swing.JFrame;
7 import javax.swing.JLabel;
8 import javax.swing.JPanel;
9
10 public class notgetball extends JFrame{
11
12     JButton pic;
13
14     public notgetball() {
15         super("NOT GET BALL!");
16         setSize(400, 400);
17         setLocation(200, 50);
18         setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
19         setVisible(true);
20
21         JPanel jp = new JPanel();
22         jp.setBackground(Color.WHITE);
23
24         jp.setLayout(new FlowLayout(FlowLayout.CENTER));
25         JLabel label = new JLabel("<html><body><p align=\"center\">這裡沒有精靈球了！</p></body></html>");
26         label.setSize(200, 200);
27         label.setFont(new java.awt.Font("Microsoft JhengHei", 1, 25));
28         jp.add(label);
29
30         ImageIcon get_ball = new ImageIcon("images\\zoro.gif");
31         pic = new JButton(get_ball);
32         pic.setSize(340, 300);
33         pic.setBackground(Color.WHITE);
34         pic.setVisible(true);
35         jp.add(pic);
36
37         this.add(jp);
38
39     }
40
41 }
```


- Backpack_map.java (地圖中背包視窗)

```
1 import java.awt.BorderLayout;
2 import java.awt.Color;
3 import java.awt.Dimension;
4 import java.awt.GridLayout;
5
6 import javax.swing.ImageIcon;
7 import javax.swing.JButton;
8 import javax.swing.JFrame;
9 import javax.swing.JLabel;
10 import javax.swing.JPanel;
11 import javax.swing.SwingConstants;
12
13 public class backpack_map extends JFrame{
14
15     JButton jb[] = new JButton[9]; //寶貝球按鈕
16     int ballnum;
17
18     public backpack_map(int bn) {
19         this.setSize(350, 350);
20         this.setLocation(536, 50);
21         this.setVisible(true);
22         this.setTitle("背包");
23
24         // 設置寶貝球
25         JPanel jp = new JPanel();
26         jp.setBackground(Color.WHITE);
27         GridLayout gl = new GridLayout(3,3);
28         jp.setLayout(gl);
29
30         String ball_path = "images/ball.jpg";
31         ImageIcon ball_icon = new ImageIcon(ball_path);
32
33         for(int i=0; i<jb.length; i++) {
34             jb[i] = new JButton();
35             jb[i].setSize(100, 100);
36             jb[i].setBackground(Color.WHITE);
37             jp.add(jb[i]);
38             if(i < (3+bn)) {
39                 jb[i].setIcon(ball_icon);
40             }
41         }
42
43         // 設置標題
44         JPanel nev_jp = new JPanel();
45         //nev_jp.setBackground(Color.WHITE);
46         nev_jp.setSize(650, 80);
47         JLabel jl = new JLabel("精靈球", SwingConstants.CENTER);
48         jl.setFont(new java.awt.Font("Microsoft JhengHei", 1, 25));
49         nev_jp.add(jl);
50
51         // 畫片排版
52         JPanel lf_jp = new JPanel();
53         JPanel ri_jp = new JPanel();
54
55         lf_jp.setPreferredSize(new Dimension(25, 350));
56         ri_jp.setPreferredSize(new Dimension(25, 350));
57
58
59         this.add(lf_jp, BorderLayout.WEST);
60         this.add(ri_jp, BorderLayout.EAST);
61         this.add(nev_jp, BorderLayout.NORTH);
62         this.add(jp, BorderLayout.CENTER);
63
64     }
65
66
67 }
```

- Card.java (地圖中卡包視窗)

```
1 import java.awt.BorderLayout;
2 import java.awt.Color;
3 import java.awt.Dimension;
4 import java.awt.GridLayout;
5 import java.awt.event.ActionEvent;
6 import java.awt.event.ActionListener;
7 import javax.swing.ImageIcon;
8 import javax.swing.JButton;
9 import javax.swing.JFrame;
10 import javax.swing.JLabel;
11 import javax.swing.JPanel;
12 import javax.swing.SwingConstants;
13
14 public class card extends JFrame implements ActionListener{
15
16     // 設置卡牌
17     JButton jb[] = new JButton[9];
18
19     public card(int g,int p,int j,int c) {
20
21         // 設置版型
22         this.setSize(426, 503);
23         this.setLocation(574, 50);
24         this.setVisible(true);
25         this.setTitle("卡牌");
26
27         // 設置放卡片的地方
28         JPanel jp = new JPanel();
29         GridLayout gl = new GridLayout(3,3);
30         jp.setLayout(gl);
31
32         for(int i=0; i<9; i++) {
33             jb[i] = new JButton();
34             jb[i].setSize(122, 151);
35             jb[i].setBackground(Color.WHITE);
36             jp.add(jb[i]);
37         }
38
39         // 皮卡丘卡牌
40         String card_pikachu_path = "images/card_pikachu.png";
41         ImageIcon card_pikachu_icon = new ImageIcon(card_pikachu_path);
42         jb[0].setIcon(card_pikachu_icon);
43         jb[0].addActionListener(this);
44
45         // 伊布卡牌
46         String card_Eevee_path = "images/card_Eevee.png";
47         ImageIcon card_Eevee_icon = new ImageIcon(card_Eevee_path);
48         jb[1].setIcon(card_Eevee_icon);
49         jb[1].addActionListener(this);
50
51         // 胖丁卡牌
52         String card_Jigglypuff_path = "images/card_Jigglypuff.png";
53         ImageIcon card_Jigglypuff_icon = new ImageIcon(card_Jigglypuff_path);
54         jb[2].setIcon(card_Jigglypuff_icon);
55         jb[2].addActionListener(this);
56     }
57 }
```

```

57 // 基拉祈卡牌
58 String card_Jirachi_path = "images/card_Jirachi.png";
59 ImageIcon card_Jirachi_icon = new ImageIcon(card_Jirachi_path);
60 jb[3].setIcon(card_Jirachi_icon);
61 jb[3].addActionListener(this);
62
63 // 如果收服到耿鬼
64 if(g == 1) {
65     String card_Gengar_path = "images/card_Gengar.png";
66     ImageIcon card_Gengar_icon = new ImageIcon(card_Gengar_path);
67     jb[4].setIcon(card_Gengar_icon);
68     jb[4].addActionListener(this);
69 }
70
71 // 如果收服到妙挖種子
72 if(p == 1) {
73     String card_Bulbasaur_path = "images/card_Bulbasaur.png";
74     ImageIcon card_Bulbasaur_icon = new ImageIcon(card_Bulbasaur_path);
75     jb[5].setIcon(card_Bulbasaur_icon);
76     jb[5].addActionListener(this);
77 }
78
79 // 如果收服到可達鴨
80 if(j == 1) {
81     String card_Psyduck_path = "images/card_Psyduck.png";
82     ImageIcon card_Psyduck_icon = new ImageIcon(card_Psyduck_path);
83     jb[6].setIcon(card_Psyduck_icon);
84     jb[6].addActionListener(this);
85 }
86
87 // 如果收服到小火龍
88 if(c == 1) {
89     String card_Charmander_path = "images/card_Charmander.png";
90     ImageIcon card_Charmander_icon = new ImageIcon(card_Charmander_path);
91     jb[7].setIcon(card_Charmander_icon);
92     jb[7].addActionListener(this);
93 }
94
95 // 放上標題
96 JPanel nev_jp = new JPanel();
97 JLabel jl = new JLabel("寶可夢卡牌", SwingConstants.CENTER);
98 jl.setFont(new java.awt.Font("Microsoft JhengHei", 1, 25));
99 nev_jp.add(jl);
100
101 // 畫片排版
102 JPanel lf_jp = new JPanel();
103 JPanel ri_jp = new JPanel();
104
105 lf_jp.setPreferredSize(new Dimension(30, 500));
106 ri_jp.setPreferredSize(new Dimension(30, 500));
107
108 this.add(lf_jp, BorderLayout.WEST);
109 this.add(ri_jp, BorderLayout.EAST);
110 this.add(nev_jp, BorderLayout.NORTH);
111 this.add(jp, BorderLayout.CENTER);
112 }

```

```

115 @Override
116 public void actionPerformed(ActionEvent arg0) {
117
118     JButton source_jb = (JButton) arg0.getSource();
119
120     // 點擊卡牌查看其角色設定
121     if(source_jb == jb[0]) {
122         pikachu cc = new pikachu();
123     } else if(source_jb == jb[1]) {
124         eevee cc = new eevee();
125     } else if(source_jb == jb[2]) {
126         jigglypuff cc = new jigglypuff();
127     } else if(source_jb == jb[3]) {
128         jilachi cc = new jilachi();
129     } else if(source_jb == jb[4]) {
130         gengar cc = new gengar();
131     } else if(source_jb == jb[5]) {
132         bulbasaur cc = new bulbasaur();
133     } else if(source_jb == jb[6]) {
134         psyduck cc = new psyduck();
135     } else if(source_jb == jb[7]) {
136         charmander cc = new charmander();
137     }
138 }
139 }

```

- Pikachu.java (精靈角色設定介紹視窗)

(另有eevee, jigglypuff, jilachi, gengar, bulbasaur, psyduck, charmander · 僅角色介紹的文字不同 · 因此僅貼上pikachu作為代表)

```
1 import javax.swing.*;
2 import java.awt.*;
3 import java.awt.event.*;
4
5 // 角色皮卡丘介紹視窗
6 class pikachu extends JFrame{
7
8     public pikachu() {
9         super("皮卡丘");
10        setSize(239, 220);
11        setLocation(200, 50);
12        setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
13        setVisible(true);
14
15        JPanel jp = new JPanel();
16        jp.setLayout(new FlowLayout(FlowLayout.CENTER));
17
18        JLabel label=new JLabel("<html><body><p align='center'>皮卡丘<br/>屬性：電<br/>HP：3<br/>攻擊：4<br/>防禦：3<br/>速度：6</p></body></html>");
19        label.setSize(200,200);
20        label.setLocation(0, 0);
21        label.setFont(new java.awt.Font("Microsoft JhengHei", 1, 20));
22        jp.add(label);
23
24        this.add(jp);
25    }
26 }
```