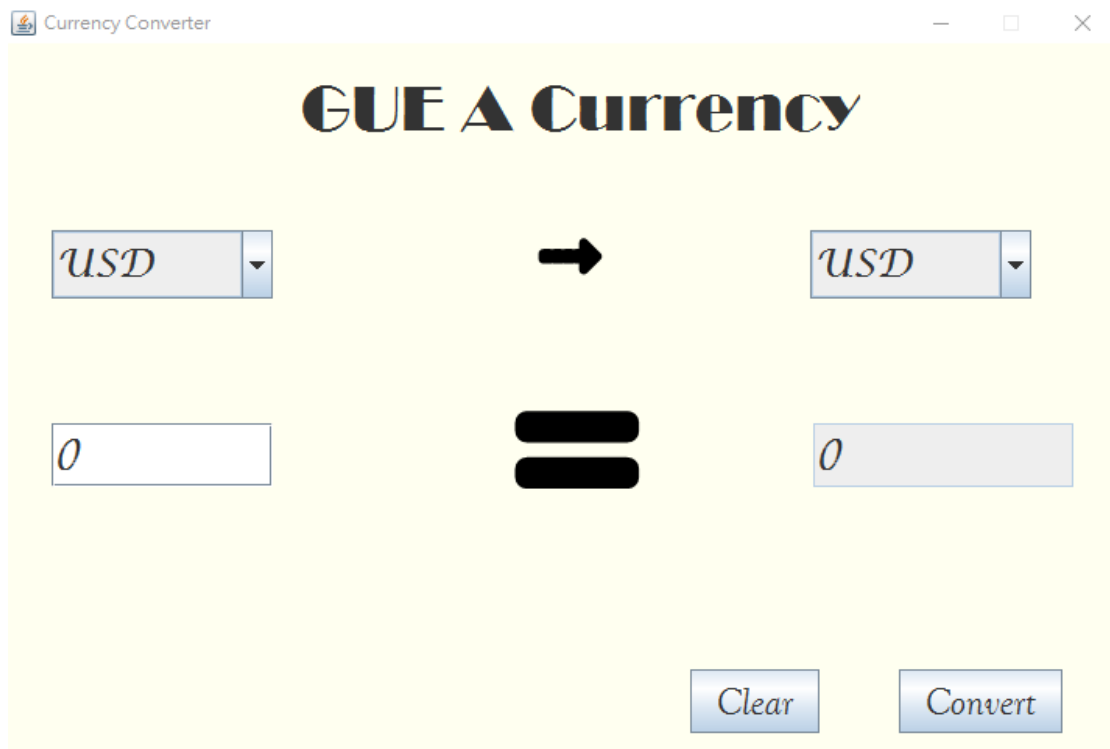


# Final Project

## GUE A Currency (NO.9)



The screenshot shows a window titled "Currency Converter" with a yellow background. At the top, the text "GUE A Currency" is displayed in a bold, black, serif font. Below this, there are two currency selection dropdown menus, both currently set to "USD". Between these dropdowns is a right-pointing arrow. Below the dropdowns are two text input fields, both containing the number "0". Between these input fields is a large equals sign. At the bottom right of the window, there are two buttons: "Clear" and "Convert".

Project number: 404530007 歐陽嘉琳

404530025 鍾茗薇

404530032 丘旖瑄

404530043 林怡廷

## ☆ 問題討論與解決方案

### 1. 貨幣金額不可小於零以及輸入不可為字母

解決方案：

```
T1 = new JTextField();
T1.setFont(new Font("Vivaldi", Font.BOLD, 30));
T1.setText("0");
T1.setColumns(10);

T1.addKeyListener(new KeyListener() { //指定只可以輸入數字和小數點
    public void keyTyped(KeyEvent e) {

        int temp = e.getKeyChar(); //讀取輸入字元 字元的常數
        System.out.println(temp); //輸出輸入字元 字元的常數
        if(temp == 10){ //限制輸入字10是Enter

        }else if(temp != 46){ // 46代表是
            if(temp != 8){ //8是Backspace

                if(temp > 57){ //字元的常數, 48~57是0~9
                    e.consume();
                }else if(temp < 48){
                    e.consume();
                }
            }
        }
    }
}
```

### 2. 同個貨幣不可執行

解決方案：

跳出警告視窗, 指示使用者重新輸入。

```
int x,y;
x=CB1.getSelectedIndex();
y=CB2.getSelectedIndex();
//如果沒有輸入任何值, 出現提醒
if (T1.getText().equals("")){
    JOptionPane.showMessageDialog( null, "請輸入值~!!");
}
else{
    double s = Double.parseDouble(T1.getText());

    //如果兩個幣種相同,出現警告
    if (x==y){
        JOptionPane.showMessageDialog( null, "兩個幣種不能相同");
    }
    //判斷幣種並進行運算
    if (x==0 && y == 1){
        T2.setText(String.valueOf(df.format(s/1.24)));
    }

    else if (x==0 && y == 2){
        T2.setText(String.valueOf(df.format(s/1.64)));
    }
}
```

### 3. 美工改善

解決方案：

利用此方法，將自己圖片顯示在 JAVA 中。

```
CB2 = new JComboBox();
CB2.setFont(new Font("Lucida Calligraphy", Font.PLAIN, 25));
CB2.setModel(new DefaultComboBoxModel(new String[] { "USD", "EUR", "GBP", "MOP" }));

CB1 = new JComboBox();
CB1.setFont(new Font("Lucida Calligraphy", Font.PLAIN, 25));
CB1.setModel(new DefaultComboBoxModel(new String[] { "USD", "EUR", "GBP", "MOP" }));

JLabel lblPic = new JLabel("");
lblPic.setIcon(new ImageIcon(CurrencyConverter.class.getResource("/Re/1.png"));
```

### 4. 使用者介面鎖定

問題: 避免視窗裡的物件，隨視窗移動而跟著移動。

解決方案：

利用此方法，鎖定畫面使畫面無法縮放。

```
public static void main(String[] args) {
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                CurrencyConverter frame = new CurrencyConverter();
                frame.setVisible(true);
                frame.setResizable(false);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}
```

## 5. 版面控制

問題: 假設轉換完金額數字太長

解決方案:

用此方法定輸入格式，此小數點五位數以後數字不會顯示。

```
public void actionPerformed(ActionEvent e) {  
    //控制為尾數不為0的5位小數的格式  
    java.text.DecimalFormat df =new java.text.DecimalFormat("#.#####");  
    int x,y;  
    x=CB1.getSelectedIndex();  
    y=CB2.getSelectedIndex();
```

## ☆ 程式中的方法

```
java.text.DecimalFormat df = new java.text.DecimalFormat("#.###00");
```

宣告了一個 DecimalFormat 物件，並可以在宣告時帶入要格式化的格式，若不帶入參數，格式規則和 NumberFormat 相同。

**\*\*格式化的字串中 0 代表一定要有值，#則代表不一定要有值**

```
frame.setResizable(false);
```

用來設定視窗可否被縮放。沒有回傳值，需要一個 boolean 型態的非物件常數當參數。

```
public void keyTyped(KeyEvent e)
```

KeyListener 介面的 keyTyped() 方法用來設定鍵盤按鍵按住與釋放間的動作。

```
int temp = e.getKeyChar();
```

KeyEvent 類別的 getKeyChar() 方法用來取得鍵盤按鍵的字元。

分工表	
歐陽嘉琳	程式規劃 程式編碼 程式 Debug
鍾茗薇	程式規劃 程式 Debug
丘旖璿	程式規劃 上台報告 PTT
林怡廷	程式規劃 文書整合處理