

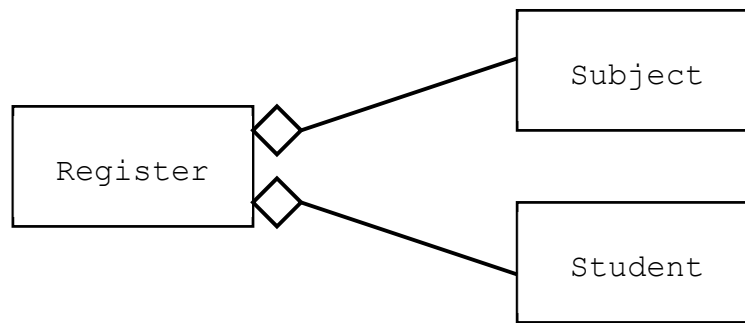
บทที่ 9 การออกแบบ Class แบบ Composition และใช้งาน Garbage Collection

การออกแบบและสร้าง Class แบบ Composition เพื่อเก็บข้อมูลการลงทะเบียน โดยมีการออกแบบ Class ตาม Class Diagram ดังนี้

Subject
- SubjectCode : String - SubjectName : String - SubjectCredit : int
+ Subject() + Subject(String Code, String Name, int Credit) + setSubjectCode(String Code) : void + setSubjectName(String Name) : void + setSubjectCredit(int Credit): void + getSubjectCode() : String + getSubjectName() : String + getSubjectCredit() : int + toString() : String

Student
- StudentCode : String - StudentName : String - StudentSurName : String
+ Student() + Student(String Code, String Name, String SurName) + setStudentCode(String Code) : void + setStudentName(String Name) : void + setStudentSurName(String SurName): void + getStudentCode() : String + getStudentName() : String + getStudentSurName() : String + toString() : String

Register
- std : Student - sub[] : Subject - max : int - count : int = -1
+ Register() + Register(Student std , int n) - createSubject(int n) : void + setStudent(Student std) : void + setSubject(Subject sub) : void + setStudentSurName(String SurName): void + setSubject(Subject sub, int n) : void + getStudent() : String + getSubject(int n) : String



การทดลองที่ 9-1

```
// File Name : Subject.java
public class Subject {
    private String SubjectCode;
    private String SubjectName;
    private int SubjectCredit;

    /** Creates a new instance of Subject */
    public Subject() {
        setSubjectCode("");
        setSubjectName("");
        setSubjectCredit(0);
    }

    public Subject(String Code, String Name, int Credit) {
        setSubjectCode(Code);
        setSubjectName(Name);
        setSubjectCredit(Credit);
    }

    public void setSubjectCode(String Code) {
        SubjectCode = Code;
    }

    public void setSubjectName(String Name) {
        SubjectName = Name;
    }

    public void setSubjectCredit(int Credit) {
        SubjectCredit = Credit;
    }

    public String getSubjectCode() {
        return (SubjectCode);
    }

    public String getSubjectName() {
        return (SubjectName);
    }
}
```

```

    public int getSubjectCredit() {
        return(SubjectCredit);
    }

    public String toString() {
        String str = "";
        str = getSubjectCode()+" "+getSubjectName()+" ";
        str += getSubjectCredit();
        return(str);
    }
}

```

```

// File Name : Student.java

public class Student {
    String StudentCode;
    String StudentName;
    String StudentSurName;

    /** Creates a new instance of Student */
    public Student() {
        setStudentCode("");
        setStudentName("");
        setStudentSurName("");
    }

    public Student(String Code, String Name, String SurName) {
        setStudentCode(Code);
        setStudentName(Name);
        setStudentSurName(SurName);
    }

    public void setStudentCode(String Code) {
        StudentCode = Code;
    }

    public void setStudentName(String Name) {
        StudentName = Name;
    }

    public void setStudentSurName(String SurName) {
        StudentSurName = SurName;
    }

    public String getStudentCode() {
        return(StudentCode);
    }

    public String getStudentName() {
        return(StudentName);
    }
}

```

```

    public String getStudentSurName() {
        return(StudentSurName);
    }

    public String toString() {
        String str = "";
        str = getStudentCode()+" "+getStudentName()+" ";
        str += getStudentSurName();
        return(str);
    }
}

```

```

// File Name : Register.java

public class Register {
    private Student std;
    private Subject sub[];
    private int max, count = -1;

    /** Creates a new instance of Register */
    public Register() {
        std = new Student();
        max = 0;
    }

    public Register(Student std, int n) {
        this.std = std;
        max = n;
        createSubject(max);
    }

    private void createSubject(int n) {
        sub = new Subject[n];
    }

    public void setStudent(Student std) {
        this.std = std;
    }

    public void setSubject(Subject sub) {
        this.sub[++count] = sub;
    }

    public void setSubject(Subject sub, int n) {
        this.sub[n] = sub;
    }

    public String getStudent() {
        return(std.toString());
    }
}

```

```
public String getSubject(int n) {  
    return(sub[n].toString());  
}  
}
```

```
// File Name : Lab9_1.java  
  
import java.awt.*;  
import java.awt.event.*;  
import javax.swing.*;  
public final class Lab9_1 extends JFrame implements ActionListener  
{  
    Student std[];  
    Subject sub[];  
    String[] studentStr, subjectStr;  
    Container container;  
    JComboBox<String> stdCombo, subCombo;  
    JLabel stdLabel, subLabel;  
    JButton addstdBtn, addsubBtn, saveBtn, clearBtn, cancelBtn;  
    JTextField stdText;  
    JTextArea subTextArea;  
    JScrollPane subScroll;  
    Register reg;  
    int count = 0;  
  
    /** Creates a new instance of Lab9_1 */  
    public Lab9_1() {  
        super("Program Student Register");  
        initStudent();  
        initSubject();  
        initGui();  
        reg = new Register(new Student(), 5);  
    }  
  
    public void initStudent() {  
        std = new Student[3];  
        studentStr = new String[3];  
        std[0] = new Student("5066260010", "AAAAA", "BBBBBBBBBBBBB");  
        std[1] = new Student("5066260024", "DDDDD", "GGGGGGGGGGG");  
        std[2] = new Student("5066260035", "HHHHH", "KKKKKKKKKKK");  
        studentStr[0] = std[0].toString();  
        studentStr[1] = std[1].toString();  
        studentStr[2] = std[2].toString();  
    }  
  
    public void initSubject() {  
        sub = new Subject[5];  
        subjectStr = new String[5];  
        sub[0] = new Subject("662305", "IT Laboratory II", 1);  
        sub[1] = new Subject("662309", "Data Structure", 3);  
        sub[2] = new Subject("662310", "Database System", 3);  
        sub[3] = new Subject("662317", "Data Communication", 3);  
        sub[4] = new Subject("662327",  
                                "Advanced Computer Programming", 3);  
    }  
}
```

```

        subjectStr[0] = sub[0].toString();
        subjectStr[1] = sub[1].toString();
        subjectStr[2] = sub[2].toString();
        subjectStr[3] = sub[3].toString();
        subjectStr[4] = sub[4].toString();
    }

    public void initGui() {
        container = getContentPane();
        container.setLayout(new FlowLayout());

        stdLabel = new JLabel("Select Student : ");
        container.add(stdLabel);
        stdCombo = new JComboBox<String>( studentStr );
        stdCombo.setMaximumRowCount(3);

        container.add(stdCombo);
        addstdBtn = new JButton("Add Student");
        addstdBtn.addActionListener(this);
        container.add(addstdBtn);
        stdText = new JTextField(40);
        stdText.setEditable(false);
        container.add(stdText);

        subLabel = new JLabel("Select Subject : ");
        container.add(subLabel);
        subCombo = new JComboBox<String>( subjectStr );
        subCombo.setMaximumRowCount(5);
        container.add(subCombo);
        addsubBtn = new JButton("Add Subject");
        addsubBtn.setEnabled(false);
        addsubBtn.addActionListener(this);
        container.add(addsubBtn);
        subTextArea = new JTextArea(5,50);
        subTextArea.setEditable(false);
        subScrollPane = new JScrollPane(subTextArea);
        container.add(subScrollPane);

        saveBtn = new JButton(" Save ");
        saveBtn.setEnabled(false);
        saveBtn.addActionListener(this);
        container.add(saveBtn);

        cancelBtn = new JButton(" Cancele ");
        cancelBtn.setEnabled(false);
        cancelBtn.addActionListener(this);
        container.add(cancelBtn);
        setSize(450,250);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
    }

    public void actionPerformed(ActionEvent event) {
        if (event.getSource() == addstdBtn) {

```

```

        int n = stdCombo.getSelectedIndex();
        stdText.setText(std[n].toString());
        reg.setStudent(std[n]);
        addstdBtn.setEnabled(false);
        addsubBtn.setEnabled(true);
        saveBtn.setEnabled(true);
        cancelBtn.setEnabled(true);
    }
    else if (event.getSource() == addsubBtn) {
        int n = subCombo.getSelectedIndex();
        subTextArea.append(sub[n].toString()+"\n");
        reg.setSubject(sub[n], count);
        count++;
        if (count == 5 ) addsubBtn.setEnabled(false);
    }
    else if (event.getSource() == saveBtn) {
        String output="";
        output = "Student :" + reg.getStudent();
        output += "\nSubject:\n";
        for(int n = 0 ; n < count; n++)
            output += reg.getSubject(n) + "\n";
        JOptionPane.showMessageDialog(this, output,
            "Registration Data",JOptionPane.INFORMATION_MESSAGE);
        resetBtn();
    }
    else if (event.getSource() == cancelBtn) {
        stdText.setText("");
        subTextArea.setText("");
        count = 0;
        resetBtn();
    }
}

public void resetBtn() {
    addstdBtn.setEnabled(true);
    addsubBtn.setEnabled(false);
    saveBtn.setEnabled(false);
    cancelBtn.setEnabled(false);
    stdText.setText("");
    subTextArea.setText("");
}

public static void main(String[] args) {

    Lab9_1 lab9_1 = new Lab9_1();
}
}

```

ผลลัพธ์

Program Student Register

Select Student : 5066260010 AAAAA BBBB BBBBBBBBBB Add Student

Select Subject : 662305 IT Laboratory II 1 Add Subject

Save Cancle

การทดลองที่ 9-2

SubjectNew
<ul style="list-style-type: none">- SubjectCode : String- SubjectName : String- SubjectCredit : int- static count : int
<ul style="list-style-type: none">+ SubjectNew()+ SubjectNew(String Code, String Name, int Credit)+ SubjectNew(SubjectNew sub)+ setSubjectCode(String Code) : void+ setSubjectName(String Name) : void+ setSubjectCredit(int Credit): void+ getSubjectCode() : String+ getSubjectName() : String+ getSubjectCredit() : int# finalize() : void+ static getCount() : int+ toString() : String


```
// File Name : SubjectNew.java
public class SubjectNew {
    private String SubjectCode;
    private String SubjectName;
    private int SubjectCredit;
    private static int count = 0;

    /** Creates a new instance of Subject */
    public SubjectNew() {
        setSubjectCode("");
        setSubjectName("");
        setSubjectCredit(0);
        count++;
    }

    public SubjectNew(String Code, String Name, int Credit) {
        setSubjectCode(Code);
        setSubjectName(Name);
        setSubjectCredit(Credit);
        count++;
    }

    public SubjectNew( SubjectNew sub) {
        setSubjectCode(sub.getSubjectCode());
        setSubjectName(sub.getSubjectName());
        setSubjectCredit(sub.getSubjectCredit());
        count++;
    }

    public void setSubjectCode(String Code) {
        SubjectCode = Code;
    }

    public void setSubjectName(String Name) {
        SubjectName = Name;
    }

    public void setSubjectCredit(int Credit) {
        SubjectCredit = Credit;
    }

    public String getSubjectCode() {
        return (SubjectCode);
    }

    public String getSubjectName() {
        return(SubjectName);
    }

    public int getSubjectCredit() {
        return(SubjectCredit);
    }

    protected void finalize() {
```

```

        count--;
    }

    public static int getCount() {
        return(count);
    }

    public String toString() {
        String str = "";
        str = getSubjectCode()+" "+getSubjectName()+" ";
        str += getSubjectCredit();
        return(str);
    }
}

```

// File Name : Lab9_2.java

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class Lab9_2 extends JFrame implements ActionListener {
    SubjectNew sub[];
    Container container;
    //JComboBox<String> stdCombo, subCombo;
    JLabel codeLabel, nameLabel, creditLabel;
    JButton addBtn, editBtn, deleteBtn, showBtn, clearBtn;
    JTextField codeText, nameText, creditText, statusText;
    JTextArea subTextArea;
    JScrollPane subScroll;

    /** Creates a new instance of Lab9_2 */
    public Lab9_2() {
        initGui();
        sub = new SubjectNew[10];
        statusText.setText("Number Object : "+ SubjectNew.getCount());
    }

    public Lab9_2(int max) {
        initGui();
        sub = new SubjectNew[max];
        statusText.setText("Number Object : "+ SubjectNew.getCount());
    }

    public void initGui() {
        container = getContentPane();
        container.setLayout(new FlowLayout());

        codeLabel = new JLabel(" Subject Code : ");
        container.add(codeLabel);
        codeText = new JTextField(10);
    }
}

```

```

        container.add(codeText);
        container.add(new JLabel("                "));

        nameLabel = new JLabel(" Subject Name : ");
        container.add(nameLabel);
        nameText = new JTextField(20);
        container.add(nameText);

        creditLabel = new JLabel("Subject Credit : ");
        container.add(creditLabel);
        creditText = new JTextField(5);
        container.add(creditText);
        container.add(new JLabel("                "));

        addBtn = new JButton("Add");
        addBtn.addActionListener(this);
        container.add(addBtn);

        editBtn = new JButton("Edit");
        editBtn.addActionListener(this);
        container.add(editBtn);

        deleteBtn = new JButton("Delete");
        deleteBtn.addActionListener(this);
        container.add(deleteBtn);

        showBtn = new JButton("Show");
        showBtn.addActionListener(this);
        container.add(showBtn);

        clearBtn = new JButton("Clear");
        clearBtn.addActionListener(this);
        container.add(clearBtn);
        subTextArea = new JTextArea(8,25);
        subTextArea.setEditable(false);
        subScroll = new JScrollPane(subTextArea);
        container.add(subScroll);

        statusText = new JTextField(30);
        statusText.setEnabled(false);
        container.add(statusText);

        setSize(500,200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
    }

    public void actionPerformed(ActionEvent event) {
        if (event.getSource() == addBtn) {
            if (SubjectNew.getCount() == sub.length)
            {
                JOptionPane.showMessageDialog( this,

```

```

        "Array full , can not add",
        "Message", JOptionPane.INFORMATION_MESSAGE);
        return;
    }
    int pos = CheckArrayEmpty();
    int n = Integer.parseInt(creditText.getText());
    sub[ pos ] = new SubjectNew( codeText.getText(),
                                nameText.getText(), n);
    subTextArea.setText( readString( sub ) );
    JOptionPane.showMessageDialog(this, "Add Subject already",
    "Message", JOptionPane.INFORMATION_MESSAGE);
    clearTextField();
}
else if (event.getSource() == editBtn) {
    String s = codeText.getText();
    int n = searchSubject( sub, s);
    if (n >= 0 ) {
        sub[n].setSubjectName(nameText.getText());
        sub[n].setSubjectCredit( Integer.parseInt(
                                creditText.getText() ) );
        subTextArea.setText( readString(sub) );
        JOptionPane.showMessageDialog(this,
                                "Edit Subject already",
        "Message", JOptionPane.INFORMATION_MESSAGE);
        clearTextField();
    }
    else {
        JOptionPane.showMessageDialog(this,
                                "can not found subject code",
        "Error Message", JOptionPane.ERROR_MESSAGE);
    }
}
else if (event.getSource() == deleteBtn) {
    String s = codeText.getText();
    int n = searchSubject(sub, s);
    if (n >= 0 ) {
        nameText.setText( sub[n].getSubjectName() );
        creditText.setText( sub[n].getSubjectCredit() + "" );
        int ans = JOptionPane.showConfirmDialog(this,
        "Delete subject ",
        "Confirm", JOptionPane.YES_NO_OPTION);
        // 0 - Yes, 1 - No
        if (ans == 0) {
            sub[n] = null;
            System.gc();
            subTextArea.setText( readString(sub) );
        }
        clearTextField();
    }
    else {
        JOptionPane.showMessageDialog(this,
        "can not found subject code",

```

```

        "Error Message", JOptionPane.ERROR_MESSAGE);
    }
}
else if (event.getSource() == showBtn) {
    String s = codeText.getText();
    int n = searchSubject(sub, s);
    if (n >= 0 ) {
        nameText.setText( sub[n].getSubjectName() );
        creditText.setText( sub[n].getSubjectCredit() + "" );
    }
    else {
        JOptionPane.showMessageDialog(this,
            "can not found subject code",
            "Error Message", JOptionPane.ERROR_MESSAGE);
    }
}
else if (event.getSource() == clearBtn) {
    clearTextField();
}
statusText.setText("Number Object : "+ SubjectNew.getCount());
}

public int CheckArrayEmpty() {
    for(int n = 0 ; n < sub.length ; n++)
        if (sub[n] == null) return( n );
    return( -1 );
}

public void clearTextField() {
    codeText.setText("");
    nameText.setText("");
    creditText.setText("");
}

public int searchSubject(SubjectNew sub[], String s) {
    for(int n= 0 ; n < sub.length ; n++) {
        if ( sub[n] != null )
            if ( s.equals(sub[n].getSubjectCode() ) )
                return(n);
    }
    return(-1);
}

public String readString(SubjectNew sub[]) {
    String str="";
    for(int n = 0 ; n < sub.length ; n++) {
        if (sub[n] != null)
            str += sub[n].toString() + "\n";
    }
    return(str);
}
}

```

```
public static void main(String[] args) {  
  
    Lab9_2 lab9_2 = new Lab9_2(15);  
}  
  
}
```

ผลลัพธ์

