

A Simple Java Example

In this simple information system, the user inputs a student's information to the program in different steps. After the information is complete, the system will display this information all together.

1) We need a class to store the student's information; so we define a new class in our project and write the following fields into it.

```
public class Student {
    private String givenName;
    private String familyName;
    private long studentNo;
    private long birthyear;

    public Student(String first, String family,long ID, long birthyear) {
        givenName=first;
        familyName=family;
        studentNo=ID;           // The first constructor
        this.birthyear=birthyear;
    }

    public Student()
    {                           // The second constructor
    }

    public long calculate_age(int current_year){
        return (current_year-birthyear);
    }

    public String make_description()
    {
        return ("Given name: " + givenName + "\n"+ "Family name: "
        + familyName + "\n" +"ID number:" + studentNo + "\n" +
        "Year of birth: " + birthyear + "\n" );
    }
}
```

2) Also we need a class for displaying the student information. So we create another class in the project and write the followings:

```
import javax.swing.JOptionPane;

public class display_info {
    public static int myear=2007;
    public static void main(String[] args) {
        String name,lname;
        long stdno,stdbirthyear;
```

```

name = JOptionPane.showInputDialog( "Enter the student first name" );
lname = JOptionPane.showInputDialog( "Enter the student last name" );
// Input the information
String stdnotemp=JOptionPane.showInputDialog( "Enter the student number" );
String stdbirthyeartemp=JOptionPane.showInputDialog("Enter the student birth year" );

stdno=Integer.parseInt(stdnotemp);
stdbirthyear=Integer.parseInt(stdbirthyeartemp);

Student s;
s=new Student(name,lname,stdno,stdbirthyear);

JOptionPane.showMessageDialog( null, "The student information is"+ "\n" +
s.make_description()+"age="+
s.calculate_age(myyear),"",JOptionPane.PLAIN_MESSAGE );

    }
}

```

1) Create the workspace and project

- a) File -> New -> Project -> Java Project
- b) Specify the Project Name
- c) Press finish

2) Add a Class to the project

- a) Press right click on the project name
- b) New -> Class
- c) Specify the class name

3) Run the program

- a) Run -> Run as -> SWT Application

4) Debug a program

- a) Run-> Debug
- b) Set Breakpoint(s): Right Click on the left side of the desired line(s).
- c) Use **Step Info** and **Step Over** for controlling the running sequence
- d) Use **Terminate** button for finishing the debug mode

