

Chức năng của các hàm trong Project

<i>STT</i>	<i>Package.Class</i>	<i>Method</i>	<i>Description</i>
1	<i>sServer.Server extends Thread</i>	<i>main()</i>	Kết nối TCP, khởi tạo Thread Pool, khởi tạo WorkerThread.
2	<i>sServer.WorkerThread implements Runnable</i>	<i>public void run()</i>	Server đọc file .java gửi lên, so sánh với input, output cho trước và gửi lại kết quả cho Client
		<i>public void writeFileJava(byte[] data, String fileName)</i>	Lưu lại file .java trong package Exercise
		<i>public Class<?> compileToClassFile(String fileName)</i>	Biên dịch file .java sang file .class
		<i>public String test(byte[] data, String className, String inputtxt, String resulttxt)</i>	Kiểm tra thiết kế và kết quả của Client.
3	<i>sServer.CheckOutput</i>	<i>public CheckOutput(String className)</i>	Khởi tạo với tên class cần kiểm tra output
		<i>public boolean check(String InputDir, String ResultDir)</i>	Dùng ProcessBuider chạy file .java với đầu vào input.txt và đầu ra so sánh với result.txt. Trả về đúng nếu giống nhau và ngược lại
4	<i>sServer.ConvertToXML</i>	<i>public ConvertToXML(String classname, Class<?> aclass)</i>	Khởi tạo với tên class cần kiểm tra thiết kế và đối tượng kiểu Class của nó.
		<i>public void convertToXML(Class aclass)</i>	Lấy các thành phần trong đối tượng Class như class, method, variable để xuất ra cây cấu trúc xml
5	<i>sServer.JClasss</i>	<i>Public String toString()</i>	Tạo các thẻ <class> với thuộc tính name, modifier, extends, implements
6	<i>sServer.JVariable</i>	<i>Public String toString()</i>	Tạo các thẻ <variable> với thuộc tính name, modifier, type
7	<i>sServer.JMethod</i>	<i>Public String toString()</i>	Tạo các thẻ <method> với thuộc tính name, modifier, type, parameter_type
8	<i>sServerMyClassLoader extends ClassLoader</i>	<i>public Class<?> defineClass(String fullClassName, byte[] data)</i>	Trả về đối tượng kiểu Class sử dụng ClassLoader.defineClass(fullClassName, data, 0, data.length)
9	<i>sClient.SocketClient</i>	<i>public SocketClient(String host, int port)</i>	Khởi tạo socket phía client
		<i>public byte[] toByteArray(File submitFile)</i>	Lưu file .java sang dạng byte[]
		<i>public String send(String exerciseName, byte[] data, int length)</i>	Gửi server tên class bài nộp, byte[] của file, độ dài byte[].

10	<i>sClient.ClientFrame</i>	<i>private void btnBrowseActionPerformed(java .awt.event.ActionEvent evt)</i>	<i>Chọn file .java bài nộp</i>
		<i>private void btnSubmitActionPerformed(java. awt.event.ActionEvent evt)</i>	<i>Tạo SocketClient gửi bài lên server, kết quả trả về lưu vào labelResult</i>
11	<i>SoSanhXML.SoSanh</i>	<i>public SoSanh(String XmlDesign, String XMLTest)</i>	<i>So sánh mã nguồn java đã được convert thành XML và file XML của bản thiết kế</i>
		<i>public boolean check()</i>	<i>Kết quả đúng sai của public SoSanh(String XmlDesign, String XMLTest)</i>
		<i>public void inDesign()</i>	<i>Đọc file XML của bản thiết kế</i>
		<i>public void inTest()</i>	<i>Đọc file XML của mã nguồn java</i>
12	<i>model.Method</i>		<i>Model của thẻ method trong XML</i>
13	<i>model.Variable</i>		<i>Model của thẻ variable trong XML</i>
14	<i>model.XMLClass</i>		<i>Model của thẻ class trong XML</i>
15	<i>exercise</i>		<i>Chứa file .java, file .class bài nộp sau khi nhận được từ server</i>
16	<i>fileTXT</i>		<i>Chứa file input và result để kiểm tra bài nộp</i>
17	<i>xmlDesign</i>		<i>Chứa file xml thiết kế mẫu</i>
18	<i>xmlTest</i>		<i>Chứa file xml sau khi được phân tích từ bai làm</i>