### 2、结对编程的小组采用测试框架 对自己“结对编程”实验的程序模块（类）进行单元测试

**1）源码**

1 package sizeyusuan;

2

3 import java.util.\*;

4 import java.io.BufferedReader;

5 import java.io.BufferedWriter;

6 import java.io.File;

7 import java.io.FileOutputStream;

8 import java.io.FileReader;

9 import java.io.FileWriter;

10 import java.io.IOException;

11 import java.io.OutputStreamWriter;

12 import java.io.PrintWriter;

13 import java.io.RandomAccessFile;

14 public class xiaoxuesheng {

15 private static Random random = new Random();

16 public static int range;

17 public static String reductionofFraction(int a, int b) {// 分数约分，用于计算结果

18 int y = 1;

19 for (int i = a; i >= 1; i--) {

20 if (a % i == 0 && b % i == 0) {

21 y = i;

22 break;

23 }

24 }

25 int z = a / y;// 分子

26 int m = b / y;// 分母

27 if (z == 0) {

28 return "0";

29 }

30 if(m==1) return z+"";

31 else return biaodashi(z,m);

32

33 }

34 public static String biaodashi(int a,int b) {//判断假分数，并化假分数为带分数

35 if(a>=b) {

36 int c;

37 c=a/b;

38 int d;

39 d=a%b;

40 {if(d==0) {return c+"";}

41 return c+"'"+d+"/"+b;}

42 }return a+"/"+b;

43 }

44

45 public static void main(String[] args){

46 Scanner sc= new Scanner(System.in);

47 System.out.println("请输入产生几以内的数字：");

48 range=sc.nextInt();

49 System.out.println("请输入产生多少个运算表达式：");

50 int num=sc.nextInt();

51 int rightcount[]=new int[num+2];

52 int wrongcount[]=new int[num+2];

53 int right1=0;

54 int wrong1=0;

55 String[] results=new String[num];int i;

56 for( i=0;i<num;i++){

57

58 String expArr[]=new String[2];//定义生成的题目

59 int a= (int) (random.nextInt(range));//分子

60 int b= (int) (random.nextInt(range));//分母

61 int c= (int) (random.nextInt(range));//另一个分子

62 int d= (int) (random.nextInt(range));//另一个分母

63 int fuhao;//运算符

64 fuhao= (int) (random.nextInt(4));

65 if(b!=0&&d!=0) {//分母均不为0时生成带有分数的计算题，同时计算结果

66 if(fuhao==0) {

67 int fenzi=a\*d+b\*c;

68 int fenmu=b\*d;

69 expArr[0]=biaodashi(a,b)+'+'+biaodashi(c,d)+'=';

70 System.out.println(expArr[0]);

71 results[i]=reductionofFraction(fenzi, fenmu);

72

73 }

74 if(fuhao==1&&a\*d-b\*c>=0) {

75 int fenzi=a\*d-b\*c;

76 int fenmu=b\*d;

77 expArr[0]=biaodashi(a,b)+'-'+biaodashi(c,d)+'=';

78 System.out.println(expArr[0]);

79 results[i]=reductionofFraction(fenzi, fenmu);

80

81 }

82 if(fuhao==1&&a\*d-b\*c<0) {

83 int fenzi=b\*c-a\*d;

84 int fenmu=b\*d;

85 expArr[0]=biaodashi(a,b)+'-'+biaodashi(c,d)+'=';

86 System.out.println(expArr[0]);

87 results[i]=reductionofFraction(fenzi, fenmu);

88

89 }

90 if(fuhao==2) {

91 int fenzi=a\*c;

92 int fenmu=b\*d;

93 expArr[0]=biaodashi(a,b)+'×'+biaodashi(c,d)+'=';

94 System.out.println(expArr[0]);

95 results[i]=reductionofFraction(fenzi, fenmu);

96

97 }

98 if(fuhao==3&&c!=0) {

99 int fenzi=a\*d;

100 int fenmu=b\*c;

101 expArr[0]=biaodashi(a,b)+'÷'+biaodashi(c,d)+'=';

102 System.out.println(expArr[0]);

103 results[i]=reductionofFraction(fenzi, fenmu);

104

105 }

106 if(fuhao==3&&c==0) {

107 break;

108 /\*c=1;

109 int fenzi=a\*d;

110 int fenmu=b\*c;

111 expArr[0]=biaodashi(a,b)+'÷'+biaodashi(c,d)+'=';

112 System.out.println(expArr[0]);

113 results[i]=reductionofFraction(fenzi, fenmu);\*/

114

115 }

116

117 }

118 else {//分母至少一个为0时生成只含有整式的运算式，同时计算结果

119 b=1; d=1;

120 if(fuhao==0) {

121 int fenzi=a\*d+b\*c;

122 int fenmu=b\*d;

123 expArr[0]=a+"+"+c+"=";

124 System.out.println(expArr[0]);

125 results[i]=reductionofFraction(fenzi, fenmu);

126

127 }

128 if(fuhao==1&&a\*d-b\*c>=0) {

129 int fenzi=a\*d-b\*c;

130 int fenmu=b\*d;

131 expArr[0]=a+"-"+c+"=";

132 System.out.println(expArr[0]);

133 results[i]=reductionofFraction(fenzi, fenmu);

134

135 }

136 if(fuhao==1&&a\*d-b\*c<0) {

137 int fenzi=b\*c-a\*d;

138 int fenmu=b\*d;

139 expArr[0]=c+"-"+a+"=";

140 System.out.println(expArr[0]);

141 results[i]=reductionofFraction(fenzi, fenmu);

142

143 }

144 if(fuhao==2) {

145 int fenzi=a\*c;

146 int fenmu=b\*d;

147 expArr[0]=c+"×"+a+"=";

148 System.out.println(expArr[0]);

149 results[i]=reductionofFraction(fenzi, fenmu);

150

151 }

152 if(fuhao==3&&c!=0) {

153 int fenzi=a\*d;

154 int fenmu=b\*c;

155 expArr[0]=a+"÷"+c+"=";

156 System.out.println(expArr[0]);

157 results[i]=reductionofFraction(fenzi, fenmu);

158

159 }

160 if(fuhao==3&&c==0) {

161 break;

162 /\*c=1;

163 int fenzi=a\*d;

164 int fenmu=b\*c;

165 expArr[0]=a+"÷"+c+"=";

166 System.out.println(expArr[0]);

167 results[i]=reductionofFraction(fenzi, fenmu);\*/

168

169 }

170

171 }

172 FileWriter fw = null;

173 try {

174

175 File f=new File("Exersies.txt");//题目写入

176 fw = new FileWriter(f, true);

177 } catch (IOException e) {

178 e.printStackTrace();

179 }if(expArr[0]!=null) {

180 PrintWriter pw = new PrintWriter(fw);

181 pw.println(i+1+"."+expArr[0]);

182 pw.flush();

183 try {

184 fw.flush();

185 pw.close();

186 fw.close();

187 } catch (IOException e) {

188 e.printStackTrace();

189 }}FileWriter fn = null;

190 try {

191

192 File f=new File("Answer.txt");//答案写入

193 fn = new FileWriter(f, true);

194 } catch (IOException e) {

195 e.printStackTrace();

196 }if(expArr[0]!=null) {

197 PrintWriter pn = new PrintWriter(fn);

198 pn.println(i+1+"."+results[i]);

199 pn.flush();

200 try {

201 fn.flush();

202 pn.close();

203 fn.close();

204 } catch (IOException e) {

205 e.printStackTrace();

206 }}

207 }

208 System.out.println("输入ok提交！");

209 Scanner sc1=new Scanner(System.in);

210 String submit=sc1.nextLine();

211 if(submit.equals("ok")){

212 String array[]=new String[num];

213 try

214 { int k=0;

215

216 FileReader fr = new FileReader("H://eclipse2//eclipse3//sizeyusuan//Your\_answers.txt");

217 BufferedReader br = new BufferedReader(fr);

218 String s ;

219 while((s = br.readLine())!=null) {//读取小学生的答案

220 array[k]=s; k++;

221 }br.close();

222 fr.close();

223 }catch(IOException e){

224 System.out.println("指定文件不存在");

225 }

226 for(int j=0;j<num;j++){

227 if(array[j].equals(results[j])) {//验证答案，统计正确和错误的个数

228

229 rightcount[j]=j+1;

230 right1++;

231 }

232 else {

233

234 wrongcount[j]=j+1;

235 wrong1++;

236 }

237 }

238 FileWriter fg = null;

239 try {

240 //反馈正确与错误题目的信息

241 File f=new File("Grade.txt");

242 fg = new FileWriter(f, true);

243 } catch (IOException e) {

244 e.printStackTrace();

245 }

246 PrintWriter pg = new PrintWriter(fg);

247 pg.println(" ");

248 pg.print("Correct:"+right1+"(");

249 for (int j = 0; j <= num; j++) {

250 if (rightcount[j] != 0) {

251 pg.print(rightcount[j] + ",");

252 }

253 }

254 pg.println(")");

255 pg.print("Wrong:"+wrong1+"(");

256 for (int j = 0; j <= num; j++) {

257 if (wrongcount[j] != 0) {

258 pg.print(wrongcount[j] + ",");

259 }

260 }

261 pg.print(")");

262 pg.flush();

263 try {

264 fg.flush();

265 pg.close();

266 fg.close();

267 } catch (IOException e) {

268 e.printStackTrace();

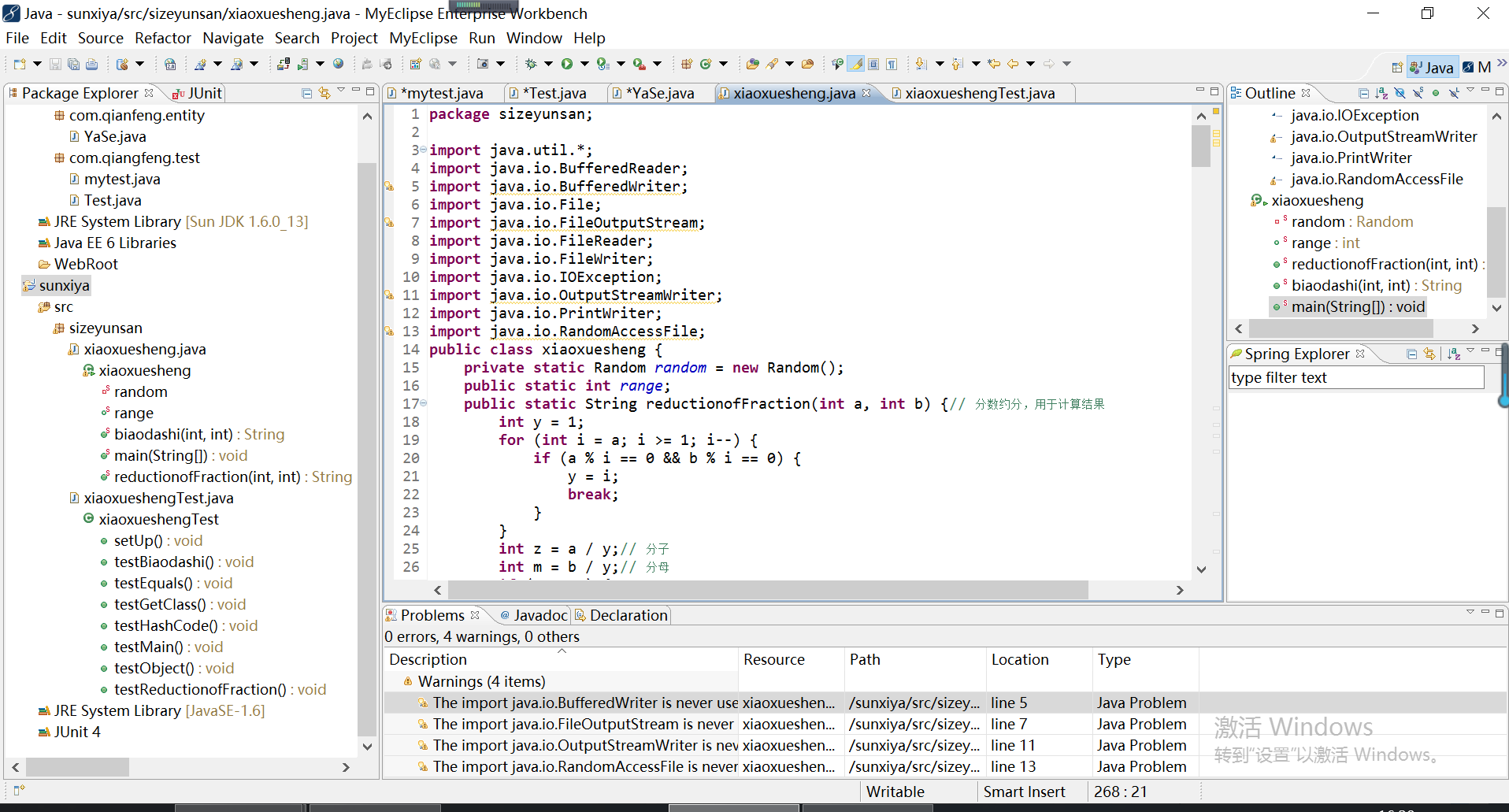
269 }}

270 }

271 }

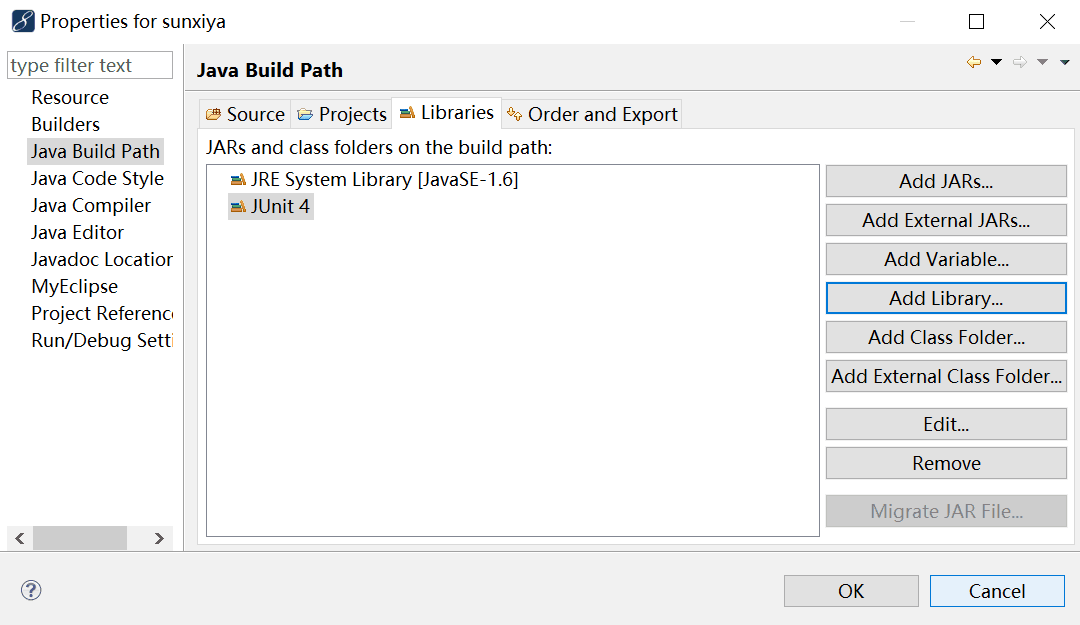
**2）测试用例设计 （结合单元测试的内容和模块功能设计测试用例）**

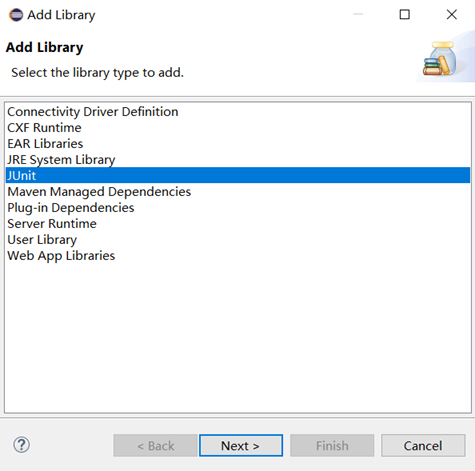
新建一个名为sunxiya 的项目，在其中编写一个 xiaoxuesheng 类。开发一个自动生成小学四则运算题目的命令行 “软件”。实现 输入你要出题的个数，随机产生四则运算，然后用户回答，并且进行打分。规则：用随机数实现100以内的加、减、乘、除运算，其中和与积不能超过100，差不为负（即须大减小），商不为小数或分数（即必须整除）。要求总计输出10个运算式，每输出一个运算式，等待输入结果，然后进行对错判断并输出。最后输出统计答对的题数与分然后对这些功能进行单元测试。



**3）安装Junit**

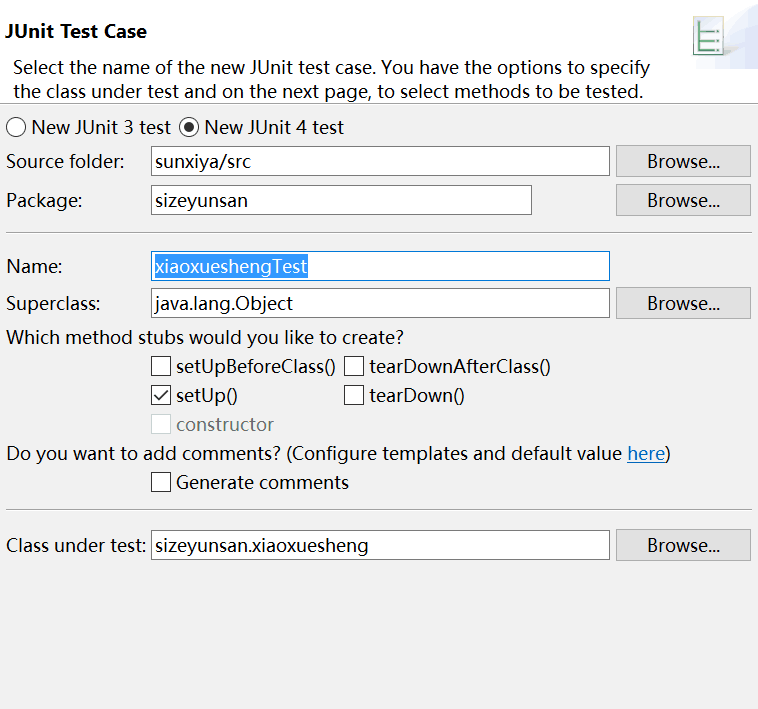
将 JUnit4单元测试包引入这个项目，在属性窗口添加Junit，选择Junit4





**4）创建测试用例**

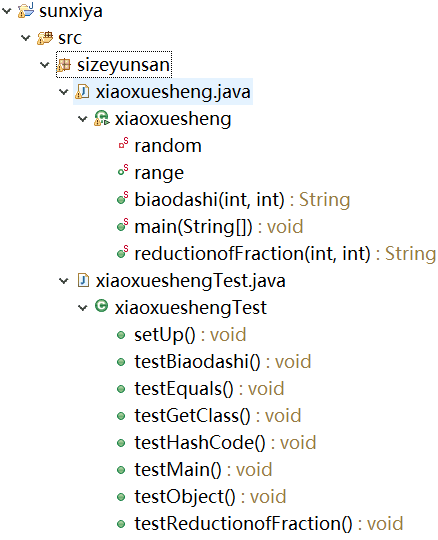
生成JUnit测试框架，在Eclipse的Package Explorer中用右键点击该类弹出菜单，选择"JUnit测试用例"。在弹出的对话框中，进行相应的选择



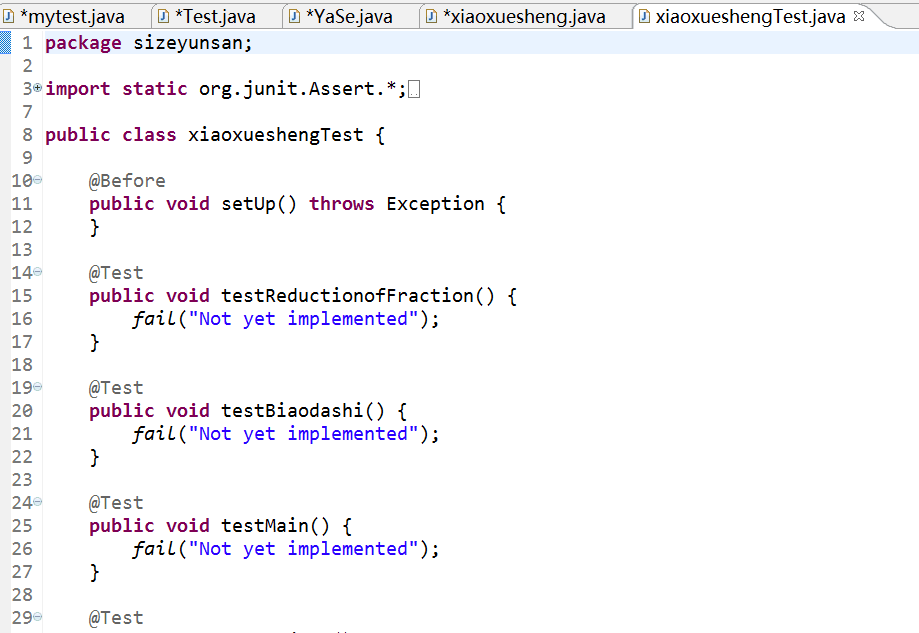
**5）测试代码**

点击“下一步”后，系统会自动列出你这个类中包含的方法，选择你要进行测试的方法。此例中，我们仅对下图所勾选的选项进行测试。

之 后 系 统 会 自 动 生 成 一 个 新 类CalculatorTest，里面包含一些空的测试用例。



**6）测试结果**



 共进行了 3个测试，其中3个测试失败

