

Resolución problema Matrices

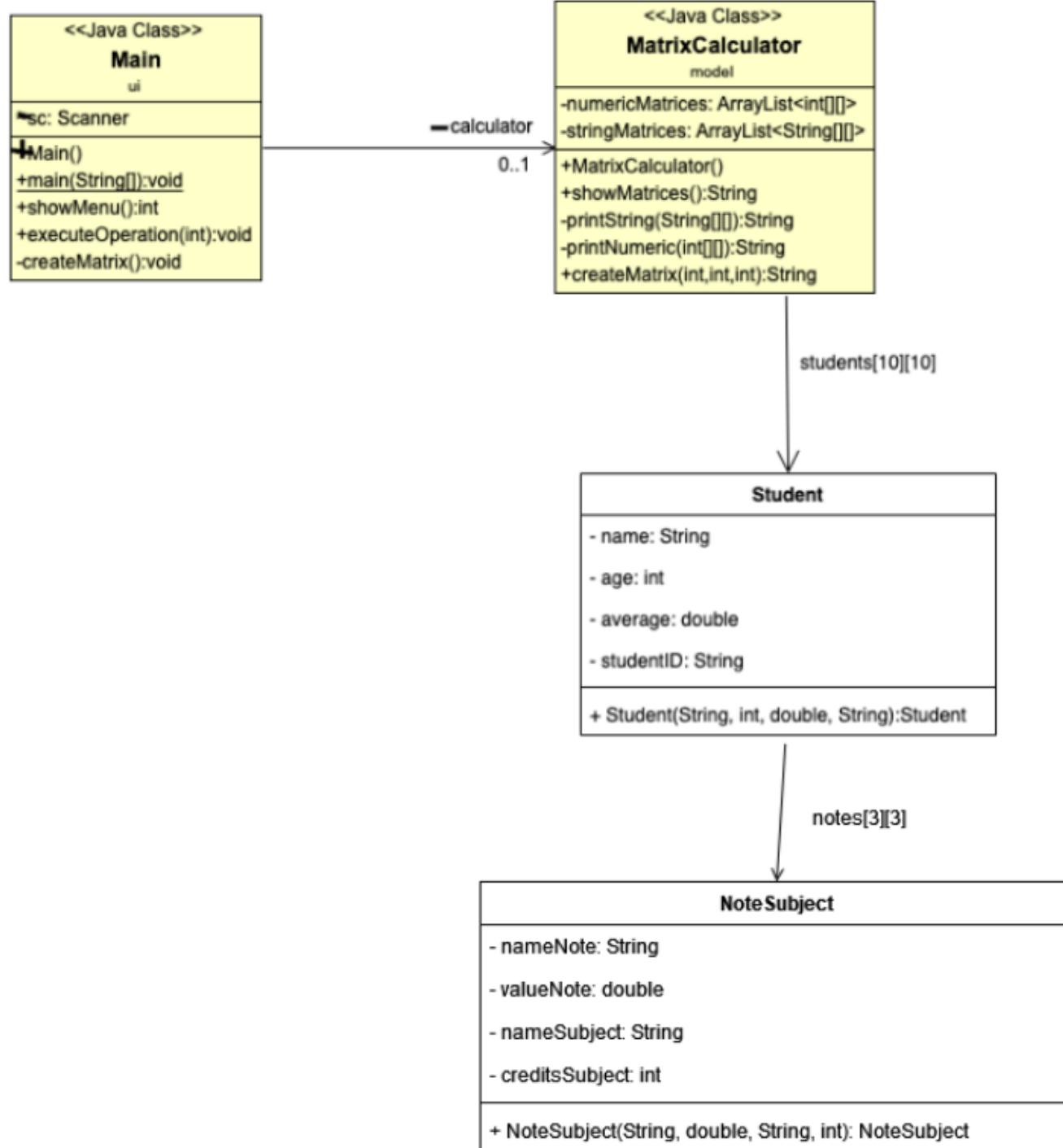
Por: Camilo Carmona Valencia

Santiago Ángel Ordoñez

Brayan Steven Ortega

Daniel Montezuma

Diego Mueses



Registro de estudiantes

- ◇ Se pidieron los datos.
- ◇ Se requirió que ningún dato tuviese mas de 18 caracteres.
- ◇ Se guardaron en una matriz de 10×10 .

```
166 public void addStudent(){
167
168     String name="",id="", holder="";
169     double average=0;
170     int age=0, number=1;
171     double[][] grades = new double[3][3];
172     double[] avgGrades = new double[3];
173
174     System.out.println(x: "What is the name of the student? (No more than 18 letters)");
175     name= sc.next();
176     sc.nextLine();
177
178     int k= name.length();
179     if (k>18){
180         do{
181
182             System.out.println(x: "The name is to long, write the name again.");
183             name= sc.next();
184             sc.nextLine();
185
186             k= name.length();
187             if(k<=18){
188                 number=0;
189             }
190
191         }while(number !=0);
192         number=1;
193     }
194
195 }
196
```

Tabla de estudiantes

```
146 public void addStudentToTable(Student student){
147     int position = FindEmptyTable();
148     if(position>0){
149         table[position][0] = "Student " + position;
150         table[position][1] = student.getName();
151         table[position][2] = student.getAge() + "";
152         table[position][3] = student.getAverage() + "";
153         table[position][4] = student.getId();
154     }
155 }
```

- ◆ Al momento del registro se añadieron los datos del estudiante a una matriz de Strings.
- ◆ En la tabla cada fila representa un estudiante y cada columna representa un atributo de este.

Impresión de la tabla

```
156 public String PrintTable(){
157     String message = " _____ \n", dataHolder = "";
158     int lengthDifference = 0;
159     for (int counter=0; counter<100&&dataHolder!=null; counter++){
160         for (int counter2=0; counter2<5&&dataHolder!=null; counter2++){
161             dataHolder = table[counter][counter2];
162             if (dataHolder!=null){
163                 lengthDifference = Math.abs(dataHolder.length()-18);
164                 for (int counter3=0; counter3<lengthDifference; counter3++){
165                     dataHolder += " ";
166                 }
167                 lengthDifference = 0;
168                 message += "|" + dataHolder + "|";
169             }
170         }
171         if (dataHolder!=null){
172             message += "\n ----- \n";
173         }
174     }
175     return message;
176 }
```

- ❖ Para imprimir la tabla se recorrieron sus filas y se añadieron sus datos a un mensaje.
- ❖ Luego este mensaje fue impreso.

La tabla

	Name	Age	Average	Student ID
Student 1	Juan	12	2.0	12234
Student 2	Pedro	13	3.0	1432
Student 3	Mariana	12	4.0	13242

Notas

- ♦ Las notas fueron registradas 9 por estudiante, 3 por materia
- ♦ Las materias fueron : Matemáticas aplicada, APO 1 e Ingeniería de Software.
- ♦ Los créditos fueron 9, 4 para matemáticas aplicadas, 3 para APO, y 2 para ingeniería de software.
- ♦ De estas notas se calcula el promedio ponderado

```
246
247
248  System.out.println(x: "Input the grades the student got\n\n1)Matematicas Aplicadas: \n");
249  for (int counter=0; counter<3; counter++){
250      System.out.println("Input the grade the student got int the exam number " + (counter+1));
251      grades[0][counter] =sc.nextDouble();
252  }
253  System.out.println(x: "\n\n2)Apo 1: \n");
254  for (int counter=0; counter<3; counter++){
255      System.out.println("Input the grade the student got int the exam number " + (counter+1));
256      grades[1][counter] =sc.nextDouble();
257  }
258  System.out.println(x: "\n\n2)ingenieria de Software: \n");
259  for (int counter=0; counter<3; counter++){
260      System.out.println("Input the grade the student got int the exam number " + (counter+1));
261      grades[2][counter] =sc.nextDouble();
262  }
263  for (int counter=0; counter<3; counter++){
264      avgGrades[counter] = (grades[counter][0] +grades[counter][1] + grades[counter][2])/3;
265  }
266  average = ((avgGrades[0]*4)+(avgGrades[1]*3)+(avgGrades[2]*2))/9;
```

Registro de notas

- ◆ Los datos que no sean el valor de la nota se llenan automáticamente dependiendo de que nota se puso.

```
121  ✓ for (int counter=0; counter<3; counter++){  
122      notes[counter][0] = new NoteSubject(("Parcial" + (counter+1)), grades[counter][0], nameSubject: "Matematicas Aplicadas", creditsSubject: 4);  
123      notes[counter][1] = new NoteSubject(("Parcial" + (counter+1)), grades[counter][1], nameSubject: "APO 1", creditsSubject: 3);  
124      notes[counter][2] = new NoteSubject(("Parcial" + (counter+1)), grades[counter][2], nameSubject: "Ingenieria de Software", creditsSubject: 2);  
125  }
```


Búsqueda de estudiantes

- Los estudiantes se buscan mediante sus Ids.
- Se hace un recorrido parcial de la matriz de estudiantes, de arriba abajo, de izquierda a derecha.

```
181  public int[] FindStudentById(String id){  
182      int[] ids = null;  
183      for (int counter=0; counter<10; counter++){  
184          for (int counter2=0; counter2<10; counter2++){  
185              if (students[counter][counter2] != null && students[counter][counter2].getId().equals(id)){  
186                  ids = new int[2];  
187                  ids[0] = counter;  
188                  ids[1] = counter2;  
189              }  
190          }  
191      }  
192      return ids;  
193  }
```

Determinación si un estudiante pasó

- ❖ Se busca al estudiante requerido.
- ❖ Se obtiene su promedio ponderado.
- ❖ Se revisa si este es mayor que 3.5.
- ❖ Se imprime su promedio y si es mayor se dice que pasó el semestre, de lo contrario se dice que perdió.

```
272 public void DidStudentAprove(){
273     String id = "";
274     int[] student = null;
275     double average = 0;
276     System.out.println(x: "input the id of the student:");
277     id = sc.next();
278     sc.nextLine();
279     student = calculator.FindStudentById(id);
280     if (student != null){
281         average = (calculator.getAverage(student[0], student[1]));
282         System.out.println("\nThe average for this student is: " + average + "\n");
283         if (average<=3.5){
284             System.out.println(x: "\nThe student falied the semester");
285         }
286         else{
287             System.out.println(x: "\nThe student passed the semester");
288         }
289     }
290 }
291 }
292 }
```

Impresión de notas

```
public String PrintGrades(){  
    String message = "  
    int lengthDifference = 0;  
    for (int counter=0; counter<3&&dataHolder!=null; counter++){  
        for (int counter2=0; counter2<4&&dataHolder!=null; counter2++){  
            if (counter2==0){  
                message += "|Parcial " + (counter+1) + "          |";  
            }  
            else{  
                dataHolder = notes[counter2-1][counter].getValueNote() + "";  
                lengthDifference = Math.abs(((dataHolder).length() - 22));  
                if (lengthDifference>0&&dataHolder!=null){  
                    for (int counter3 = 0; counter3<lengthDifference; counter3++){  
                        dataHolder += " ";  
                    }  
                }  
                message += "|" + dataHolder + "|";  
            }  
        }  
    }  
    if (dataHolder!=null){  
        message += "\n ----- \n";  
    }  
}  
return message;
```

Matematicas Aplicadas APO 1

Ingenieria de Software \n

- ❖ Se busca al estudiante.
- ❖ Se genera la tabla de notas, clasificando las columnas, y recorriendo las filas.
- ❖ Se devuelve una String con la tabla y se imprime

La tabla de notas

	Matematicas Aplicadas	APO 1	Ingenieria de Software
Parcial 1	3.0	2.3	4.3
Parcial 2	4.0	4.3	3.5
Parcial 3	3.0	5.0	4.0

Demostración