

## TRABAJO EN CLASE - TEMA 07

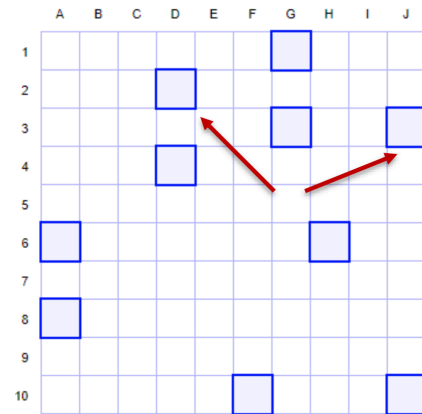
### Ejercicio 1:

- Crear un diagrama de clases de UML (buscar herramientas libres para su elaboración)
- Un programa que cree 2 objetos de la clase Soldado. El usuario ingresa los valores por teclado y se muestran los datos en formato: "nombre, nivelVida => ColFila". Usando toString()

NivelVida es aleatorio [1..5].

Soldado0, 5 => D2

Soldado1, 3 => J3



### Clase Soldado.java:

```
1  import java.util.*;
2  class Soldado {
3      private String nombre;
4      private int nivelVida;
5      private char columna;
6      private int fila;
7
8      // Constructor
9      public Soldado(String nombre, char columna, int fila) {
10         this.nombre = nombre;
11         this.columna = columna;
12         this.fila = fila;
13         this.nivelVida = generarNivelVida();
14     }
15
16     // Método para generar nivel de vida aleatorio entre 1 y 5
17     private int generarNivelVida() {
18         Random random = new Random();
19         return random.nextInt(5) + 1;
20     }
21
22     // Método toString para mostrar los datos en el formato requerido
23     @Override
24     public String toString() {
25         return nombre + ", " + nivelVida + " => " + columna + fila;
26     }
27 }
```

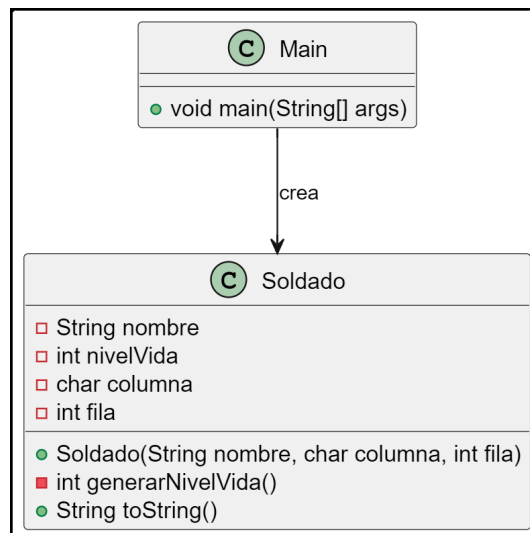
## Clase Ejercicio1.java:

```
1  import java.util.Scanner;
2
3  public class Ejercicio1 {
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6
7          // Crear el primer soldado
8          System.out.print("Ingrese el nombre del Soldado 1: ");
9          String nombre1 = sc.nextLine();
10         System.out.print("Ingrese la columna (A-J) del Soldado 1: ");
11         char columna1 = sc.next().charAt(0);
12         System.out.print("Ingrese la fila (1-10) del Soldado 1: ");
13         int fila1 = sc.nextInt();
14         Soldado soldado1 = new Soldado(nombre1, columna1, fila1);
15
16         // Crear el segundo soldado
17         System.out.print("Ingrese el nombre del Soldado 2: ");
18         sc.nextLine(); // Esta línea limpia el buffer
19         String nombre2 = sc.nextLine();
20         System.out.print("Ingrese la columna (A-J) del Soldado 2: ");
21         char columna2 = sc.next().charAt(0);
22         System.out.print("Ingrese la fila (1-10) del Soldado 2: ");
23         int fila2 = sc.nextInt();
24         Soldado soldado2 = new Soldado(nombre2, columna2, fila2);
25
26         // Mostrar los resultados
27         System.out.println(soldado1);
28         System.out.println(soldado2);
29     }
30 }
```

## Consola:

```
'-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\jhona\AppData\Roaming\Code\User\workspaces\
Ejercicio1'
● Ingrese el nombre del Soldado 1: Benjamin
  Ingrese la columna (A-J) del Soldado 1: C
  Ingrese la fila (1-10) del Soldado 1: 3
  Ingrese el nombre del Soldado 2: José
  Ingrese la columna (A-J) del Soldado 2: J
  Ingrese la fila (1-10) del Soldado 2: 8
  Benjamin, 2 => C3
  Jos?, 4 => J8
```

## Diagrama UML:



## Ejercicio 2:

- Implementar un videojuego (iterativo) en modo gráfico.
- Crear diagrama de clases y código.
- Se debe simular una guerra entre 2 reinos (ingresar sus nombres por teclado).
- La guerra está compuesta de batallas (la cantidad será aleatoria entre 5 y 20).
- Cada batalla se da entre 2 soldados, uno por cada reino. Los tipos de soldado son Arquero, Espadachín y Caballero, los cuales tienen cierto nivel de vida. Para cada batalla se generará aleatoriamente el tipo de soldado y su nivel de vida según la siguiente tabla.
- Para determinar al ganador de la batalla se debe considerar las probabilidades (balance de poder) proporcionales al nivel de vida de cada soldado.

### Ejemplo:

**R1: Arquero 5 vs R2: Espadachín 5** -> probabilidades son 50% para ambos

**R1: Arquero 4 vs R2: Caballero 8** -> probabilidades son 33,33% y 66,66%

La elección del ganador de la batalla es aleatoria pero de acuerdo a las probabilidades generadas. Cuando acabe la guerra se deberá mostrar el marcador final de las batallas ganadas y qué reino ganó la guerra (se permite empates).

Al acabar una guerra el videojuego deberá permitir empezar otra desde cero.

Tipo de Soldado	Nivel de Vida
Arquero	3 - 5
Espadachín	5 - 7
Caballero	7 - 9

### Clase Soldado.java:

```
1  import java.util.*;
2  public class Soldado {
3      private String tipo;
4      private int nivelVida;
5
6      public Soldado(String tipo, int nivelVida) {
7          this.tipo = tipo;
8          this.nivelVida = nivelVida;
9      }
10
11     public String getTipo() {
12         return tipo;
13     }
14
15     public int getNivelVida() {
16         return nivelVida;
17     }
18 }
```

```

19 // Método estático para generar un soldado aleatorio
20 public static Soldado generarSoldado(Random random) {
21     String[] tipos = {"Arquero", "Espadachín", "Caballero"};
22     String tipo = tipos[random.nextInt(tipos.length)];
23     int nivelVida = 0;
24
25     switch (tipo) {
26         case "Arquero":
27             nivelVida = random.nextInt(3) + 3;
28             break;
29         case "Espadachín":
30             nivelVida = random.nextInt(3) + 5;
31             break;
32         case "Caballero":
33             nivelVida = random.nextInt(3) + 7;
34             break;
35     }
36     return new Soldado(tipo, nivelVida);
37 }
38
39 @Override
40 public String toString() {
41     return tipo + " (Vida: " + nivelVida + ")";
42 }
43 }

```

### Clase Reino.java:

```

1 public class Reino {
2     private String nombre;
3     private int victorias;
4
5     public Reino(String nombre) {
6         this.nombre = nombre;
7         this.victorias = 0;
8     }
9
10    public String getNombre() {
11        return nombre;
12    }
13
14    public int getVictorias() {
15        return victorias;
16    }
17
18    public void incrementarVictorias() {
19        this.victorias++;
20    }
21
22    public void resetVictorias() {
23        this.victorias = 0;
24    }
25 }

```

## Clase Guerra.java:

```
1  import java.util.*;
2  public class Guerra {
3      private Reino reino1;
4      private Reino reino2;
5      private int cantidadBatallas;
6      private int victoriasReino1;
7      private int victoriasReino2;
8      private int empates;
9      private Random random;
10
11     public Guerra(Reino reino1, Reino reino2) {
12         this.reino1 = reino1;
13         this.reino2 = reino2;
14         this.random = new Random();
15         this.cantidadBatallas = random.nextInt(16) + 5; // 5-20
16         this.victoriasReino1 = 0;
17         this.victoriasReino2 = 0;
18         this.empates = 0;
19     }
20
21     public void simularGuerra() {
22         System.out.println("\n--- Inicio de la Guerra entre " + reino1.getNombre()
23                             + " y " + reino2.getNombre() + " ---");
24         System.out.println("Cantidad de batallas: " + cantidadBatallas + "\n");
25
26         for (int i = 1; i <= cantidadBatallas; i++) {
27             System.out.println("Batalla " + i + ":");
28             Soldado soldado1 = Soldado.generarSoldado(random);
29             Soldado soldado2 = Soldado.generarSoldado(random);
30
31             System.out.println(reino1.getNombre() + ": " + soldado1);
32             System.out.println(reino2.getNombre() + ": " + soldado2);
33
34             determinarGanador(soldado1, soldado2);
35
36             System.out.println();
37         }
38
39         mostrarResultados();
40     }
41 }
```

```

42 // Método para determinar el ganador de una batalla
43 private void determinarGanador(Soldado s1, Soldado s2) {
44     int vidaTotal = s1.getNivelVida() + s2.getNivelVida();
45     double probabilidadReino1 = ((double) s1.getNivelVida() / vidaTotal) * 100;
46     double probabilidadReino2 = ((double) s2.getNivelVida() / vidaTotal) * 100;
47
48     double aleatorio = random.nextDouble() * 100;
49
50     System.out.printf("Probabilidades - %s: %.2f%% | %s: %.2f%%\n",
51         reino1.getNombre(), probabilidadReino1,
52         reino2.getNombre(), probabilidadReino2);
53     System.out.printf("Número aleatorio generado: %.2f\n", aleatorio);
54
55     if (aleatorio < probabilidadReino1) {
56         System.out.println("Ganador de la batalla: " + reino1.getNombre());
57         victoriasReino1++;
58         reino1.incrementarVictorias();
59     } else if (aleatorio < probabilidadReino1 + probabilidadReino2) {
60         System.out.println("Ganador de la batalla: " + reino2.getNombre());
61         victoriasReino2++;
62         reino2.incrementarVictorias();
63     } else {
64         System.out.println("Empate en la batalla.");
65         empates++;
66     }
67 }
68
69 private void mostrarResultados() {
70     System.out.println("--- Resultados Finales ---");
71     System.out.println(reino1.getNombre() + " ganó " + victoriasReino1 + " batallas.");
72     System.out.println(reino2.getNombre() + " ganó " + victoriasReino2 + " batallas.");
73     System.out.println("Batallas empatadas: " + empates);
74
75     if (victoriasReino1 > victoriasReino2) {
76         System.out.println("¡" + reino1.getNombre() + " ha ganado la guerra!");
77     } else if (victoriasReino1 < victoriasReino2) {
78         System.out.println("¡" + reino2.getNombre() + " ha ganado la guerra!");
79     } else {
80         System.out.println("La guerra ha terminado en empate.");
81     }
82 }
83 }

```

## Clase Ejercicio2.java:

```

1 import java.util.*;
2
3 public class Ejercicio2 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         Random random = new Random();
7         boolean continuar = true;
8
9         System.out.print("=== Simulador de Guerra entre dos Reinos ===");
10
11         while (continuar) {
12             System.out.print("\nIngrese el nombre del primer reino: ");
13             String nombreReino1 = sc.nextLine().trim();
14             while (nombreReino1.isEmpty()) {
15                 System.out.print("El nombre no puede estar vacío. Ingrese nuevamente: ");
16                 nombreReino1 = sc.nextLine().trim();
17             }

```

```

18
19     System.out.print("Ingrese el nombre del segundo reino: ");
20     String nombreReino2 = sc.nextLine().trim();
21     while (nombreReino2.isEmpty()) {
22         System.out.print("El nombre no puede estar vacío. Ingrese nuevamente: ");
23         nombreReino2 = sc.nextLine().trim();
24     }
25
26     Reino reino1 = new Reino(nombreReino1);
27     Reino reino2 = new Reino(nombreReino2);
28
29     // simulación de la guerra
30     Guerra guerra = new Guerra(reino1, reino2);
31     guerra.simularGuerra();
32
33     System.out.print("\n¿Desea iniciar otra guerra? (S/N): ");
34     String respuesta = sc.nextLine().trim().toUpperCase();
35
36     while (!respuesta.equals("S") && !respuesta.equals("N")) {
37         System.out.print("Respuesta inválida. Ingrese 'S' para sí o 'N' para no: ");
38         respuesta = sc.nextLine().trim().toUpperCase();
39     }
40
41     if (respuesta.equals("N")) {
42         continuar = false;
43         System.out.println("\n¡Gracias por jugar este simulador de guerra! Hasta luego ;)");
44     }
45 }
46 }
47 }

```

## Consola:

```

s\Ejercicio_2_fd0b644f\bin' 'Ejercicio2'
=== Simulador de Guerra entre dos Reinos ===
Ingrese el nombre del primer reino: Germania
Ingrese el nombre del segundo reino: Candia

--- Inicio de la Guerra entre Germania y Candia ---
Cantidad de batallas: 11

Batalla 1:
Germania: Caballero (Vida: 8)
Candia: Arquero (Vida: 4)
Probabilidades - Germania: 66.67% | Candia: 33.33%
Número aleatorio generado: 42.07
Ganador de la batalla: Germania

Batalla 2:
Germania: Arquero (Vida: 3)
Candia: Espadachín (Vida: 5)
Probabilidades - Germania: 37.50% | Candia: 62.50%
Número aleatorio generado: 28.71
Ganador de la batalla: Germania

Batalla 3:
Germania: Arquero (Vida: 3)
Candia: Caballero (Vida: 8)
Probabilidades - Germania: 27.27% | Candia: 72.73%
Número aleatorio generado: 79.28
Ganador de la batalla: Candia

Batalla 4:
Germania: Espadachín (Vida: 5)
Candia: Caballero (Vida: 8)
Probabilidades - Germania: 38.46% | Candia: 61.54%
Número aleatorio generado: 50.86
Ganador de la batalla: Candia

Batalla 5:
Germania: Arquero (Vida: 5)
Candia: Arquero (Vida: 4)
Probabilidades - Germania: 55.56% | Candia: 44.44%
Número aleatorio generado: 16.33
Ganador de la batalla: Germania

```

```

Batalla 6:
Germania: Espadachín (Vida: 6)
Candia: Espadachín (Vida: 7)
Probabilidades - Germania: 46.15% | Candia: 53.85%
Número aleatorio generado: 53.43
Ganador de la batalla: Candia

Batalla 7:
Germania: Caballero (Vida: 7)
Candia: Caballero (Vida: 7)
Probabilidades - Germania: 50.00% | Candia: 50.00%
Número aleatorio generado: 77.17
Ganador de la batalla: Candia

Batalla 8:
Germania: Espadachín (Vida: 7)
Candia: Caballero (Vida: 9)
Probabilidades - Germania: 43.75% | Candia: 56.25%
Número aleatorio generado: 62.72
Ganador de la batalla: Candia

Batalla 9:
Germania: Espadachín (Vida: 7)
Candia: Arquero (Vida: 4)
Probabilidades - Germania: 63.64% | Candia: 36.36%
Número aleatorio generado: 33.36
Ganador de la batalla: Germania

Batalla 10:
Germania: Caballero (Vida: 9)
Candia: Arquero (Vida: 3)
Probabilidades - Germania: 75.00% | Candia: 25.00%
Número aleatorio generado: 99.01
Ganador de la batalla: Candia

Batalla 11:
Germania: Arquero (Vida: 4)
Candia: Arquero (Vida: 5)
Probabilidades - Germania: 44.44% | Candia: 55.56%

```

```

Batalla 11:
Germania: Arquero (Vida: 4)
Candia: Arquero (Vida: 5)
Probabilidades - Germania: 44.44% | Candia: 55.56%
Número aleatorio generado: 51.76
Ganador de la batalla: Candia

--- Resultados Finales ---
Germania ganó 4 batallas.
Candia ganó 7 batallas.
Batallas empatadas: 0
¡Candia ha ganado la guerra!

¿Desea iniciar otra guerra? (S/N): s

Ingrese el nombre del primer reino: Turquía
Ingrese el nombre del segundo reino: Inglaterra

--- Inicio de la Guerra entre Turquía y Inglaterra ---
Cantidad de batallas: 9

Batalla 1:
Turquía: Caballero (Vida: 9)
Inglaterra: Arquero (Vida: 4)
Probabilidades - Turquía: 69.23% | Inglaterra: 30.77%
Número aleatorio generado: 79.60
Ganador de la batalla: Inglaterra

Batalla 2:
Turquía: Caballero (Vida: 8)
Inglaterra: Arquero (Vida: 4)
Probabilidades - Turquía: 66.67% | Inglaterra: 33.33%
Número aleatorio generado: 25.29
Ganador de la batalla: Turquía

Batalla 3:
Turquía: Arquero (Vida: 3)
Inglaterra: Espadachín (Vida: 5)
Probabilidades - Turquía: 37.50% | Inglaterra: 62.50%
Número aleatorio generado: 29.27
Ganador de la batalla: Turquía

Batalla 4:
Turquía: Caballero (Vida: 8)
Inglaterra: Espadachín (Vida: 7)
Probabilidades - Turquía: 53.33% | Inglaterra: 46.67%
Número aleatorio generado: 91.94
Ganador de la batalla: Inglaterra

```

```

Batalla 5:
Turquía: Espadachín (Vida: 6)
Inglaterra: Caballero (Vida: 9)
Probabilidades - Turquía: 40.00% | Inglaterra: 60.00%
Número aleatorio generado: 92.84
Ganador de la batalla: Inglaterra

```

```

Batalla 6:
Turquía: Espadachín (Vida: 7)
Inglaterra: Arquero (Vida: 3)
Probabilidades - Turquía: 70.00% | Inglaterra: 30.00%
Número aleatorio generado: 14.04
Ganador de la batalla: Turquía

```

```

Batalla 7:
Turquía: Espadachín (Vida: 6)
Inglaterra: Espadachín (Vida: 5)
Probabilidades - Turquía: 54.55% | Inglaterra: 45.45%
Número aleatorio generado: 54.56
Ganador de la batalla: Inglaterra

```

```

Batalla 8:
Turquía: Espadachín (Vida: 5)
Inglaterra: Arquero (Vida: 3)
Probabilidades - Turquía: 62.50% | Inglaterra: 37.50%
Número aleatorio generado: 21.74
Ganador de la batalla: Turquía

```

```

Batalla 9:
Turquía: Espadachín (Vida: 6)
Inglaterra: Espadachín (Vida: 5)
Probabilidades - Turquía: 54.55% | Inglaterra: 45.45%
Número aleatorio generado: 32.98
Ganador de la batalla: Turquía

```

```

--- Resultados Finales ---
Turquía ganó 5 batallas.
Inglaterra ganó 4 batallas.
Batallas empatadas: 0
¡Turquía ha ganado la guerra!

```

```

¿Desea iniciar otra guerra? (S/N): n

```

```

--- Resultados Finales ---
Turquía ganó 5 batallas.
Inglaterra ganó 4 batallas.
Batallas empatadas: 0
¡Turquía ha ganado la guerra!

```

```

¡Gracias por jugar este simulador de guerra! Hasta luego ;)

```

```

PS: C:\Users\ibona\OneDrive\Documents\University\Universidad

```



## Diagrama UML:

