Proyecto # 1 – Fundamentos de programación – Docente: Paola Noreña

|  |  |
| --- | --- |
| **Equipo de Trabajo** | **Nombre** |
| David Espitia |
| Jesus Dominguez |
| Samuel Velez G |
| **Link github** | https://github.com/JDavidCE2022/ProyectoFundamentosDeProgramacionZombies.git |

Información general

La alcaldesa de Alexandria (comunidad de The Walking Dead), lo contrata a usted para realizar un software que permite gestionar la información de Zombies y Ubicaciones seguras de la ciudad.

La alcaldesa le brinda la siguiente información:

* Zombies: cada Zombie tiene un nombre, una salud, fecha de nacimiento, y tipo de sangre.
* Ubicaciones: cada Ubicación tiene un nombre, una distancia a Alexandria en metros (float) y una cantidad de Zombies (int) que merodean la ubicación.

Clase Principal

Implementa lo siguiente:

* Cree una clase llamada Principal.
* Cree un método de clase main desde donde iniciará toda la ejecución del proyecto.
* Cree un método mostrarMenu, el cual ejecutará un while infinito con las siguientes características:
  + Al inicio del while pídale al usuario que ingrese un número (muestre primero un mensaje donde quede claro si ingresa 0 que pasa, si ingresa 1 que pasa, etc).
  + Si se ingresa el número 0, se detiene la ejecución del while y finaliza la ejecución del programa.
  + Si se ingresa el número 1, se le piden al usuario por pantalla que ingrese nombre, salud, fecha de nacimiento, y tipo de sangre de un Zombie, y se crea una instancia de la clase Zombie. Almacene ese Zombie en un arreglo de Zombies (ese arreglo será como su base de datos de Zombies de la ciudad).
  + Si se ingresa el número 2, se esperaría mostrar por pantalla la información de todos los zombies almacenados hasta el momento.
  + Si se ingresa el número 3, se esperaría mostrar por pantalla la información de la cantidad de zombies creados hasta el momento.
  + Si se ingresa el número 4, muestre por pantalla la información de todos los zombies que tengan tipo de sangre “O+” o “AB+”.
  + Si se ingresa el número 5, muestre por pantalla la información de todos los zombies que nacieron después del 2000.
  + Si se ingresa el número 6, disminuya a la mitad la salud de todos los zombies (imagínese que se lanzó una bomba en toda la ciudad – ¿Cómo sobrevive la gente? no lo sabemos, pero sobreviven).
  + Si se ingresa el número 7, se le piden al usuario por pantalla que ingrese nombre, distancia a Alexandria y cantidad de Zombies de una ubicación. Almacene esa Ubicación en un arreglo de Ubicaciones (ese arreglo será como su base de datos de Ubicaciones de la ciudad).
  + Si se ingresa el número 8, se esperaría mostrar por pantalla la información de todas las ubicaciones almacenadas hasta el momento.
  + Si se ingresa el número 9, se esperaría mostrar por pantalla la información de la ubicación más segura de la ciudad (aquella con el menor número de Zombies).
  + Si se ingresa el número 10, se esperaría mostrar por pantalla la información de la todas las ubicaciones de la ciudad ordenadas por distancia a Alexandria. La más cercana debe aparecer de primero y la más lejana de última.
  + Si se ingresa el número 11, se esperaría mostrar por pantalla una frase aleatoria sobre Zombies (almacene en alguna parte 5 frases aleatorias sobre zombies).
  + Si se ingresa el número 12, invéntese alguna funcionalidad interesante que usted cree que seria útil para mostrar a la alcaldesa de Alexandria.
* Invoque el método mostrarMenu desde el método main, y verifique que su proyecto funciona (cree zombies, cree ubicaciones, liste zombies, liste ubicaciones, etc). Verifique que todo funciona.

Reglas importantes sobre el programa

* No haga un print, dentro de las clases Zombie o Ubicación.
* No haga uso del Scanner, dentro de las clases Zombie o Ubicación.
* Verifique que el código que usted haga si está bien dividido. Cree los métodos pertinentes en las ubicaciones (clases) pertinentes. Una forma fácil de identificar si se código va por mal camino, es si el método mostrarMenu contiene demasiado código.

Instrucciones de entrega:

* Desarrolle este proyecto en equipos de 2 o 3 personas.
* Cuando termine el proyecto (o a medida que lo vaya desarrollando), súbalo a GitHub. Para esta entrega lo único que usted le deberá enviar al docente es el enlace del proyecto GitHub (en Teams podrá encontrar un tutorial de conexión de BlueJ y GitHub).
* Defina con sus compañeros de equipo un estilo de codificación, Todos deberán usar un mismo estilo para definir atributos (por ejemplo, todos los atributos se definen con camelCase), para definir clases, para definir ciclos, variables, etc. Por ejemplo, podrían utilizar esta guía como base: <https://google.github.io/styleguide/javaguide.html>
* Se reservará un espacio para revisar el proyecto (totalmente terminado) el próximo **sábado 23 de abril**.