

Ejercicios de Strings - Java

Documentación oficial:

<https://docs.oracle.com/javase/7/docs/api/java/lang/String.html>

1. No sabía abreviar: Disponemos de una lista de nombres de personas, cada uno con su correspondiente forma de tratamiento (Doctor, Doctora, Señor, Señora, Ingeniero, Ingeniera). El problema, es que en la lista, las formas de tratamiento están escritas de forma completa, por ejemplo "Doctor Juan Pérez", "Ingeniera Juana Pérez". Utilizando los métodos de la clase string, abreviar para cada elemento de la lista la forma de tratamiento. Por ejemplo: Dr. Juan Perez, Ing. Juana Perez.

2. Texto y sistema legacy: Luego de leer un archivo con información acerca de estudiantes de la utn, nuestro programa recibe una lista de Strings que representan a los alumnos. En base a esta lista, tendremos que crear objetos del tipo Alumno con la siguientes características: Nombre, Apellido, N° Legajo, DNI, Sexo. El formato de los registros que recibimos es: MJuanXXXXXXPerezXXXXX01512340123654. Según la documentación, el formato de registro es, 1 caracter para sexo, 10 para nombre, 10 para apellido, 6 para número de legajo, 8 para el DNI.

3. Anagramas: Un anagrama del griego ἀνά -ana-, prefijo que significa "de vuelta", y γράμμα -gramma-, "letra") es una palabra o frase que resulta de la transposición de letras de otra palabra o frase. Por ejemplo:

Generador de anagramas

El generador de anagramas crea anagramas a partir de las letras dadas. A diferencia de los otros buscadores de palabras, el generador de anagramas de palabras alista las palabras que incluyen **todas las letras dadas**.

Letras

dardo

Hasta 15 símbolos.

Buscar

Palabra	Longitud
DADOR	5
DARDO	5
DORAD	5
RODAD	5

Las letras CH, LL y RR se deben escribir entre paréntesis (CH), para que el programa no lo interprete como letras independientes. Aquí tiene un ejemplo.

Crear un programa que determine si una palabra es anagrama de otra, por ejemplo, para “dardo” y “dador” el resultado debe ser “son anagramas”, para “java” y “baja” el resultado debe ser “no son anagramas”.

4. Aprender a leer: Necesitamos crear un programa que nos permita determinar si un niño que está aprendiendo a leer tiene posibilidades de cambiar una palabra por otra. Sabemos que según el nivel de lectura y la capacidad de concentrarse del niño, va a leer N letras, y de ahí en adelante el resultado es incierto. Por ejemplo: si el niño lee la cantidad de letras N=4 y luego inventa, podría confundir “computadora” con “compartir”, dado que solo lee “comp” y luego se arriesga a adivinar.

Nuestra misión es, sabiendo N y dos palabras, determinar si es probable o no que confunda las palabras. Ejemplo negativo: N=5 palabras: “paladín” y “palangana” en este caso no se puede “confundir”.

Ejercicios de StringBuilder.

Documentación oficial:

<https://docs.oracle.com/javase/7/docs/api/java/lang/StringBuilder.html>

1. Sistema de estudiantes: Tenemos un sistema que utiliza un objetos del tipo Estudiante los mismos del punto 2 de los ejercicios de string. Este ha sido creado con java, pero lamentablemente debemos interactuar con el mismo sistema legacy que nos daba registros con ese formato complicado. El principal problema, es que debemos exportar strings sin saltos de línea, es decir que nuestro resultado debe ser un string de una sola línea muy larga. Si bien java optimiza la concatenación de strings a la hora de compilar (por ejemplo, no es necesario usar StringBuilder en un toString). Cuando iteremos sobre nuestra lista de estudiantes si será necesario usar esta clase optimizada.

Crear un método que transforme nuestra lista de estudiantes en un string de una sola línea, para poder alimentar el sistema legacy.

2. Errores de tipo: Tenemos un sistema donde guardamos unos párrafos de información acerca de las alergias de los estudiantes de la facultad. Lamentablemente la persona a cargo de cargar esta información, tiene una tendencia a presionar de a varias teclas a la vez. Por alguna razón extraña, no podemos tocar todo el contenido del registro, y solo podemos eliminar un caracter por vez. Utilizando StringBuilder, tenemos que crear un método que nos permita indicar qué posición tiene el caracter que sobra y eliminarlo.

Por ejemplo: String “es alergico a loas abjhejas” numero: 4, resultado: “es alergico a loas abjhejas”

3. Estudiantes con legajo erróneo: Por un error en nuestro sistema legacy, muchos estudiantes fueron cargados con un número de legajo aleatorio. Utilizando StringBuilder, debemos reemplazar los espacios de caracteres que debería ocupar el número de legajo (que

actualmente están en: 104030, por ejemplo) por el número de legajo que recibimos por parámetro. Por ejemplo:

MJuanXXXXXXPerezXXXXX91512940123654 debe ser arreglado con el numero de legajo 066935, dando como resultado el registro: MJuanXXXXXXPerezXXXXX06693540123654

Nota: No debemos utilizar el objeto como soporte, porque los registros fallidos son muchos, y debemos manipular sólo el string de entrada.

Ejercicios de Date and Time - Java 8

Documentación:

LocalDateTime: <https://docs.oracle.com/javase/8/docs/api/java/time/LocalDateTime.html>

LocalDate: <https://docs.oracle.com/javase/8/docs/api/java/time/LocalDate.html>

LocalTime: <https://docs.oracle.com/javase/8/docs/api/java/time/LocalTime.html>

ZonedDateTime: <https://docs.oracle.com/javase/8/docs/api/java/time/ZonedDateTime.html>

ZonedId: <https://docs.oracle.com/javase/8/docs/api/java/time/ZonedId.html>

Ejercicios:

LocalDate:

1. Mostrar la fecha actual por pantalla.
2. Mostrar tu fecha de nacimiento por pantalla.
3. Utilizar LocalDate para determinar si una persona es mayor que otra.
4. Mostrar que día de la semana fué, 10 días luego de tu nacimiento.
5. Mostrar el día de tu cumpleaños para el año 2020.

LocalTime:

1. Mostrar la hora actual.
2. Mostrar qué hora sería dos horas y 40 minutos después del medio día.
3. Mostrar la hora actual en París realizando el cálculo.
4. Dadas dos horas, determinar cuál de las dos es "mas tarde"
5. A partir de ahora, tenemos 5 horas para resolver un problema de programación. Qué sucede con la hora?

LocalDateTime:

1. Mostrar hora y fecha actual.
2. Repetir la consigna del punto 5 de LocalTime, qué sucede con la fecha?
3. Calcular la hora y fecha actual de París, utilizando ZonedDateTime.
4. Buscar el próximo lunes desde la fecha actual.
5. Buscar cuál fué tu primer fin de semana.

