ÍNDICE

[Generador de código semántico 1](#_Toc182832463)

[Desarrollo de la gramática biblioteca: 1](#_Toc182832464)

[Desarrollo de las gramáticas Individuales: 3](#_Toc182832465)

[Usuario 3](#_Toc182832466)

[Generación RDF Usuario: 3](#_Toc182832467)

[Libros 5](#_Toc182832468)

[Generación RDF Libro: 5](#_Toc182832469)

[Préstamo 7](#_Toc182832470)

[Generación RDF Libro: 7](#_Toc182832471)

[RDF Validator 9](#_Toc182832472)

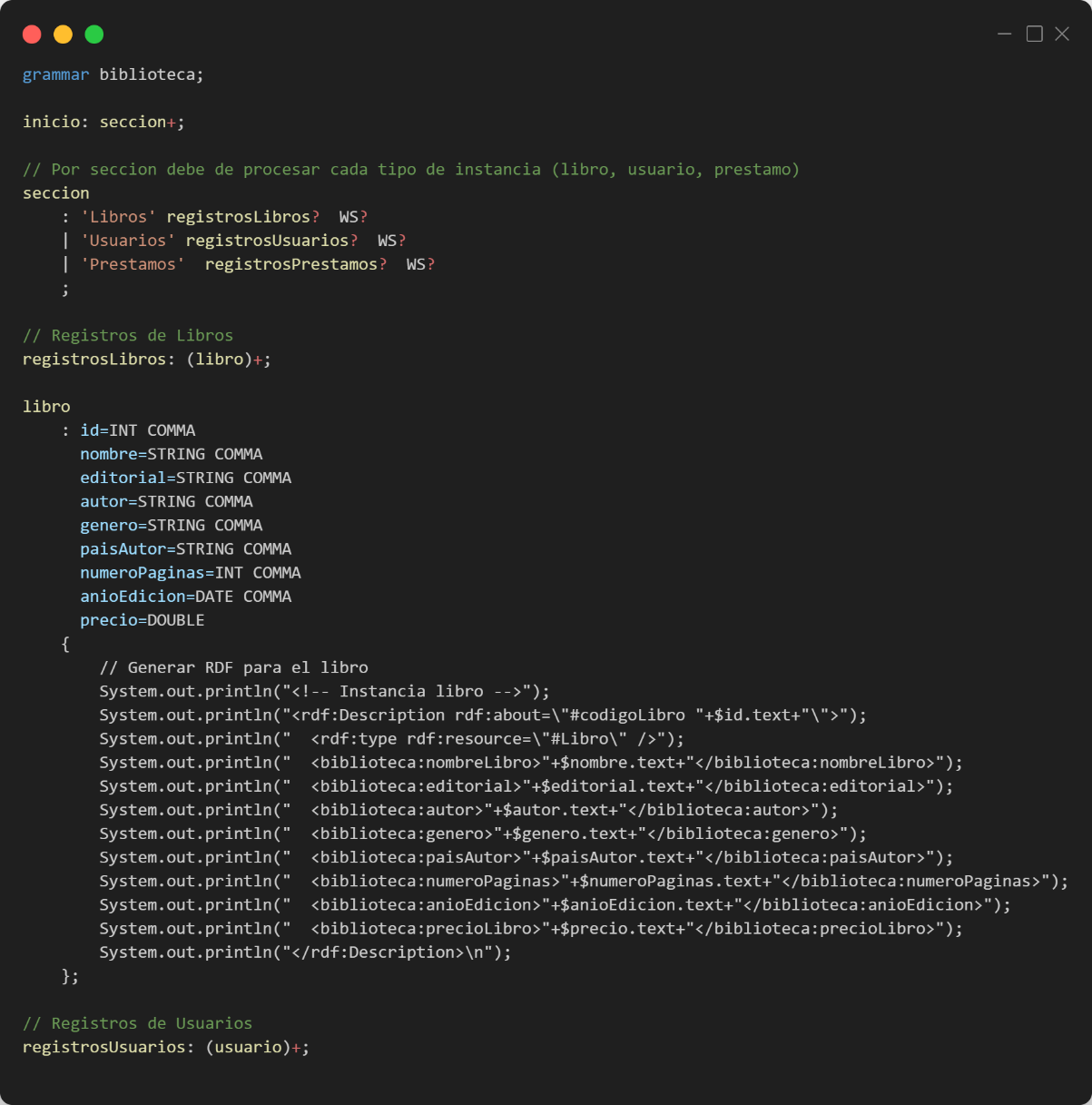
[Integración en IDE 10](#_Toc182832473)

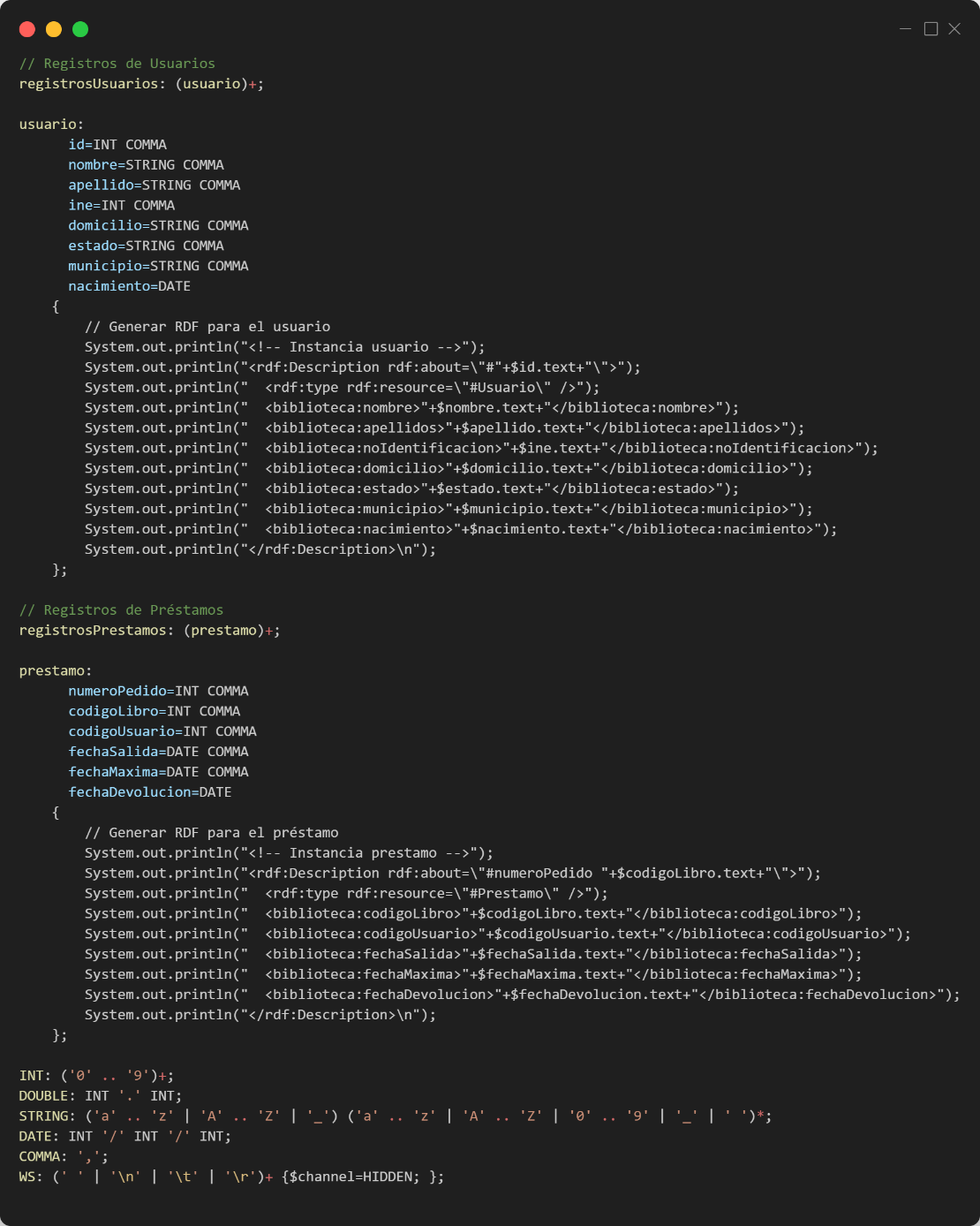
# Generador de código semántico

Para comenzar con la explicación , tenemos que empezar por definir de manera corta que es un generador de código semántico, según la biblioteca de datos de IBM podemos definir al generador de código semántico una herramienta o tecnología que ayuda a convertir código no semántico o no estructurado en código HTML semántico, es decir, en código que utiliza etiquetas y elementos HTML para describir el significado y la estructura del contenido de una página web, con esta definición podemos comenzar con el desarrollo de la actividad.

## Desarrollo de la gramática biblioteca:

Como primer paso analizaremos por completo el funcionamiento de la gramática principal, la cual destaca por tener las 3 generaciones de código RDF en el mismo archivo, el código de la gramática es el siguiente:





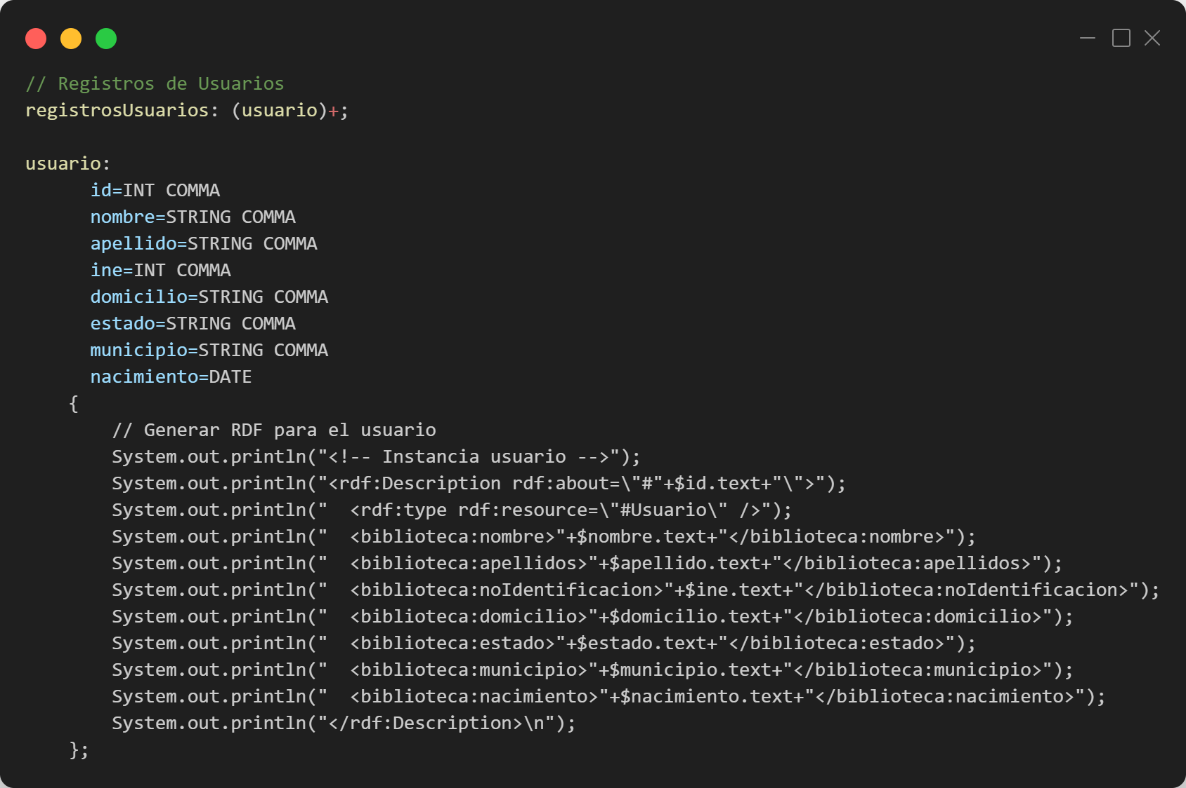
¿Como podríamos describir esta parte?, bueno primero tenemos que tener conciencia de que la gramática está diseñada para convertir registros textuales sobre libros, usuarios y préstamos en representaciones RDF que pueden ser fácilmente procesadas y consumidas en la web semántica, analizando los datos que llegan de entrada pues cuando se detecta un registro válido, se genera un bloque RDF que describe ese registro.

En donde cada registro de libros, usuarios o préstamos se traduce a un bloque RDF en formato XML con etiquetas específicas para cada tipo de dato (Cosa que veremos analizaremos más adelante).

## Desarrollo de las gramáticas Individuales:

### Usuario

Se definió una estructura para procesar información de usuarios y generar datos en formato RDF (Fig.1.0).

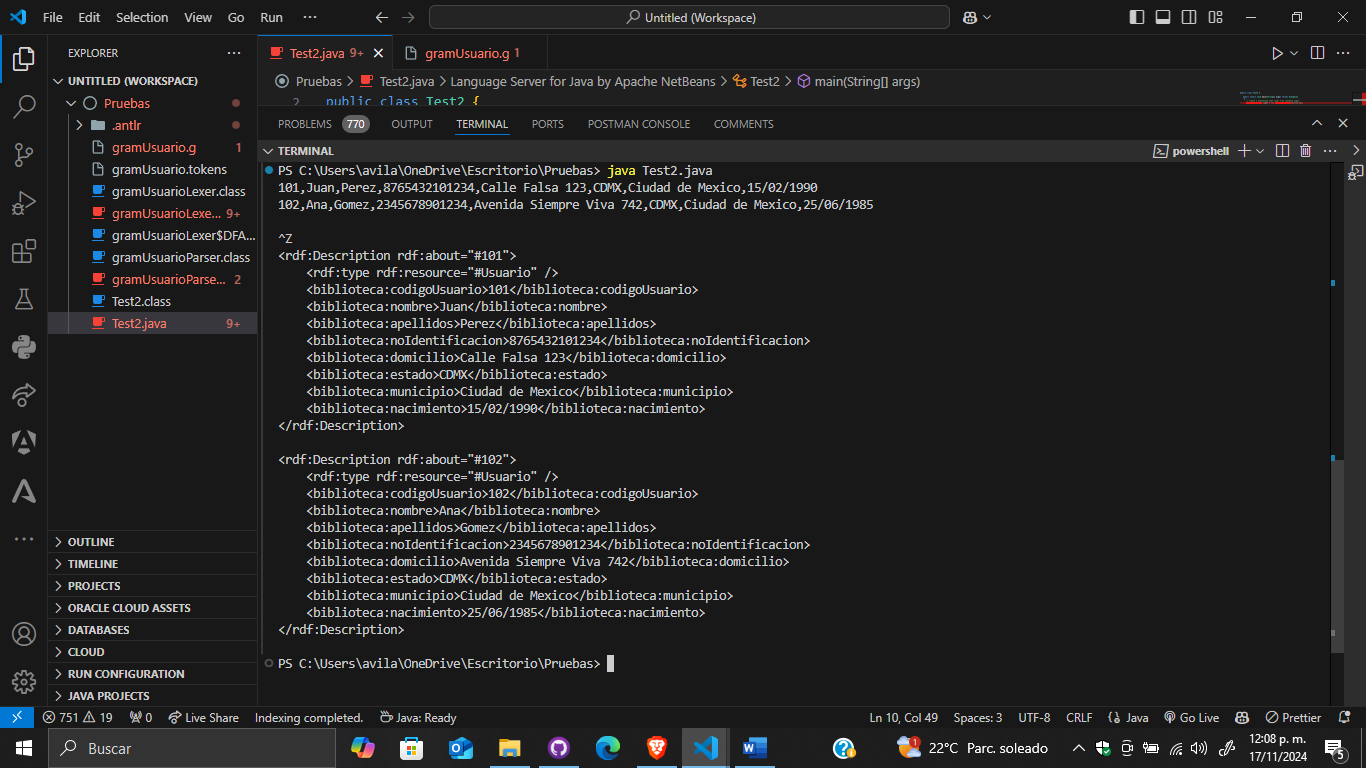


Figura

Como primera parte de la explicación del código, tenemos que estructurar la información de un usuario, en este caso usamos la estructura representada por un “ID” (el identificador único del usuario), un “NOMBRE” (El nombre del usuario), un “APELLIDO” (apellido del usuario) , el “INE” (representación de la identificación oficial del usuario), el “Domicilio”, “ESTADO” y “DOMICILIO” (representación del domicilio del usuario) y el “NACIMIENTO” (que representa la fecha de nacimiento del usuario), estos datos en general son la estructura de como tendrían que llegar los registros para poder ser convertidos a formato RDF como veremos a continuación.

### Generación RDF Usuario:

El bloque llamado “Generación RDF” es el encargado de generar el código RDF a partir de los valores capturados, esto sucede cada vez que se reconoce una instancia de usuario, permitiendo automatizar la generación de los formatos RDF de cada uno de los usuarios registrados (Fig.1.1).



(Fig.1.1)

En este caso, ¿Que podemos destacar de la ejecución de este código?, como primera parte notamos que tal como se explico en la figura anterior, la estructura de la instancia del usuario generara el código RDF del mismo, un ejemplo claro de esto es la siguiente instancia:

***101,Juan,Perez,8765432101234,Calle Falsa 123,CDMX,Ciudad de Mexico,15/02/1990***

A partir de la cual se genera el código RDF correspondiente (fig. 1.2)

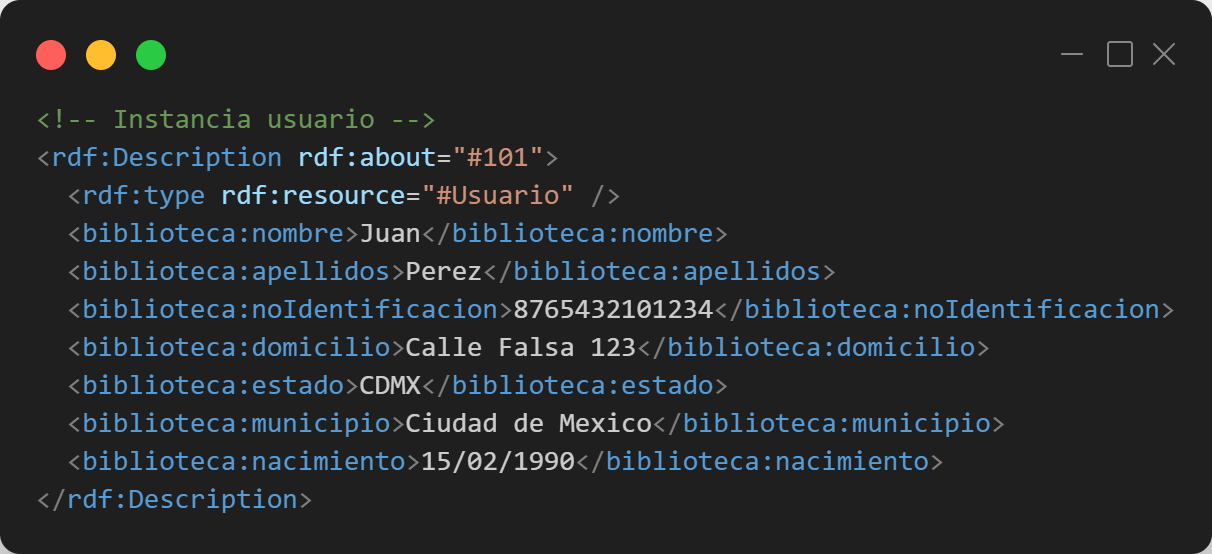
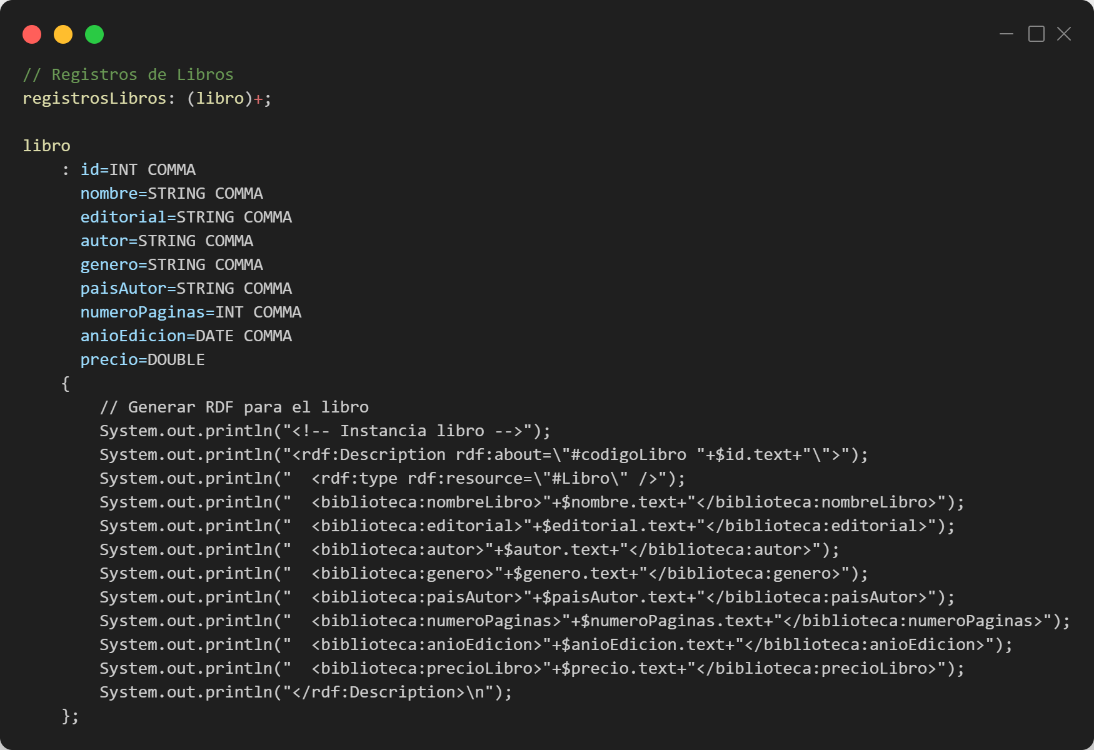


Fig. (1.2)

Al igual que en el resto de las gramáticas podemos decir que validar que cumplan con un formato específico, al cumplir con este requisito podemos transformar cada registro de usuario en una representación RDF.

### Libros

Se definió una estructura para procesar información de los libros para generar datos en formato RDF (Fig.1.3).



En esta gramática podemos ver la organización de los campos como “ID” (Identificador único del libro), el “NOMBRE, “EDITORIAL”, “AUTOR”, “GENERO”, “PAIS\_AUTOR”, “NUMERO\_PAGINAS” y “PRECIO” (datos relevantes sobre el libro), estos son los principales datos que se tendrían que verificar antes de poder generar el código RDF a partir de las instancias detectadas.

### Generación RDF Libro:

El bloque denominado "Generación RDF" se encarga de transformar automáticamente la información de los libros en un formato RDF, esto sucede cada vez que se identifica una instancia de libro, generando una representación semántica (Fig. 1.4)

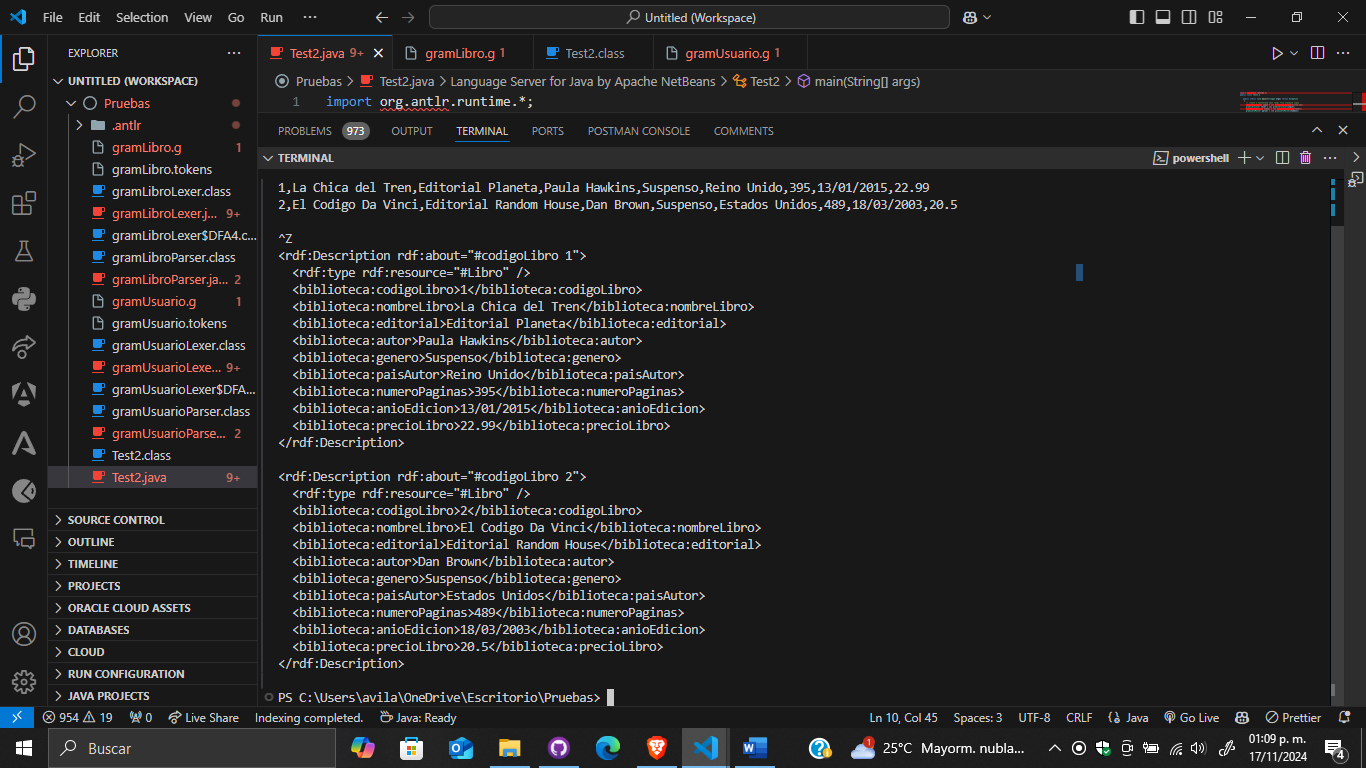


Fig. (1.4)

Al igual que en el caso de los usuarios tenemos que analizar la instancia que se usa para generar los RDF de los libros, en este caso la estructura se respeta, generando de manera correcta el RDF con la estructura esperada.

***1,La Chica del Tren,Editorial Planeta,Paula Hawkins,Suspenso,Reino Unido,395,13/01/2015,22.99***

A partir de la anterior instancia se genera el código RDF correspondiente (fig. 1.5)

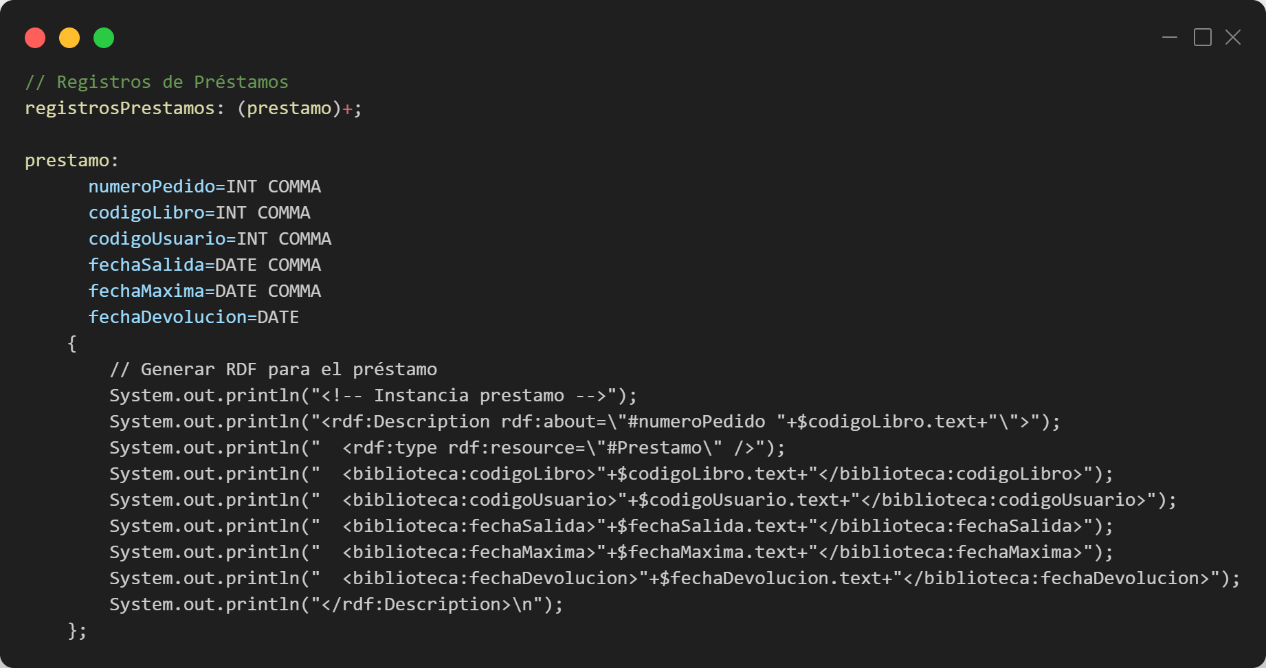


(Fig. 1.5)

Como podemos ver, la gramática del libro debe de cumplir con un formato especificado para garantizar la conversión exitosa al formato RDF, destacando también que este enfoque de automatización permite procesar múltiples registros de manera automatizada.

### Préstamo

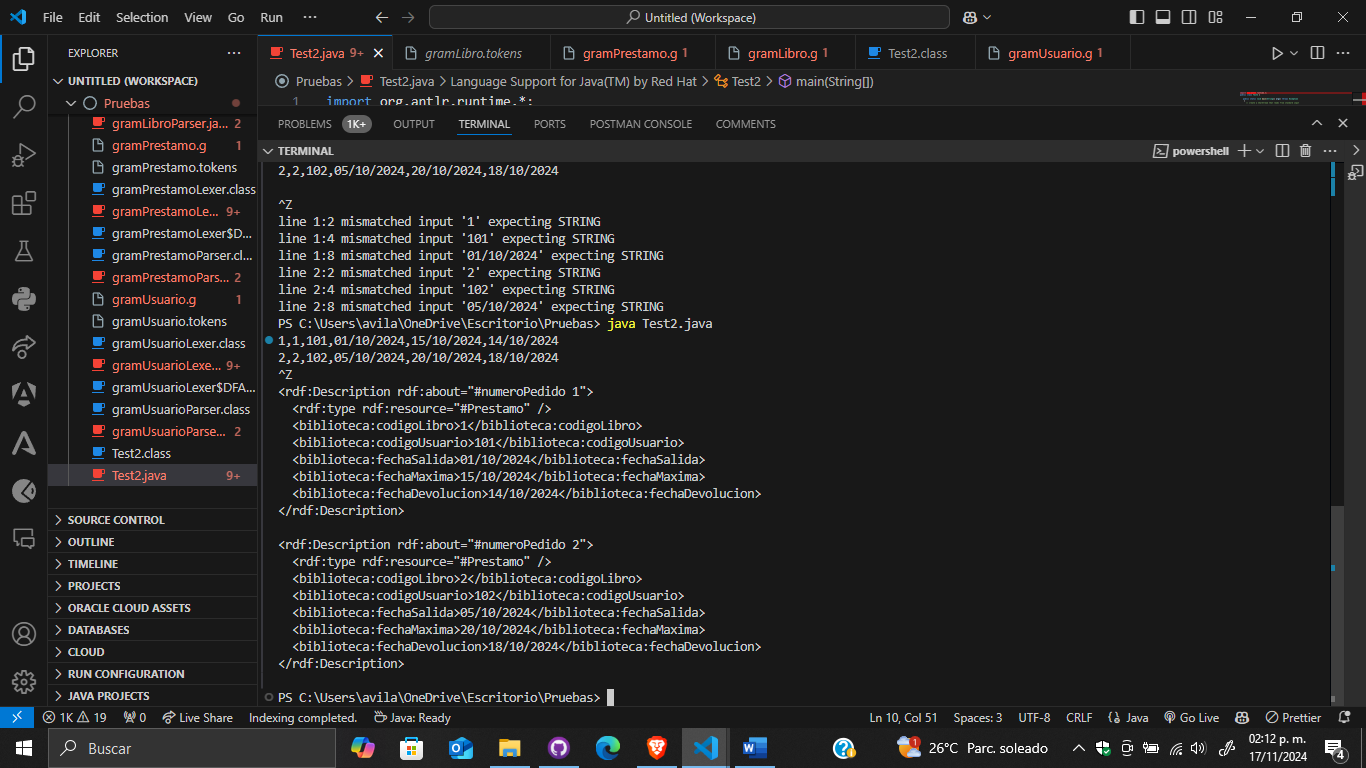
Se definió una estructura para procesar información de los libros para generar datos en formato RDF (Fig.1.6).



En este caso esta gramática es especial ya que de alguna u otra forma llega a relacionar ambas gramáticas anteriores (en cuestión de la estructura), con esto en mente podemos definir por ejemplo los requisitos para poder validar esta gramática, de los cuales son necesarios el “NUMERO DE PEDIDO”, “CODIGO DEL LIBRO”, “CODIGO DEL USUARIO”, FECHA DE SALIDA”, “FECHA MAXIMA” y “FECHA DE DEVOLUCION”, lo que nos muestra como se tiene que comportar la estructura de las instancias para la correcta generación del código RDF.

### Generación RDF Libro:

Al terminar las validaciones y el llenado del formato correcto de la instancia podemos empezar la generación del código RDF a partir de las instancias seleccionadas (Fig. 1.7)



Aquí podemos ver como se genera el código RDF a partir de la instancia del préstamo, incluyendo las propiedades relacionadas con el libro, usuario y las fechas mas relevantes del préstamo, como podemos ver en la siguiente instancia:

***1,1,101,01/10/2024,15/10/2024,14/10/2024***

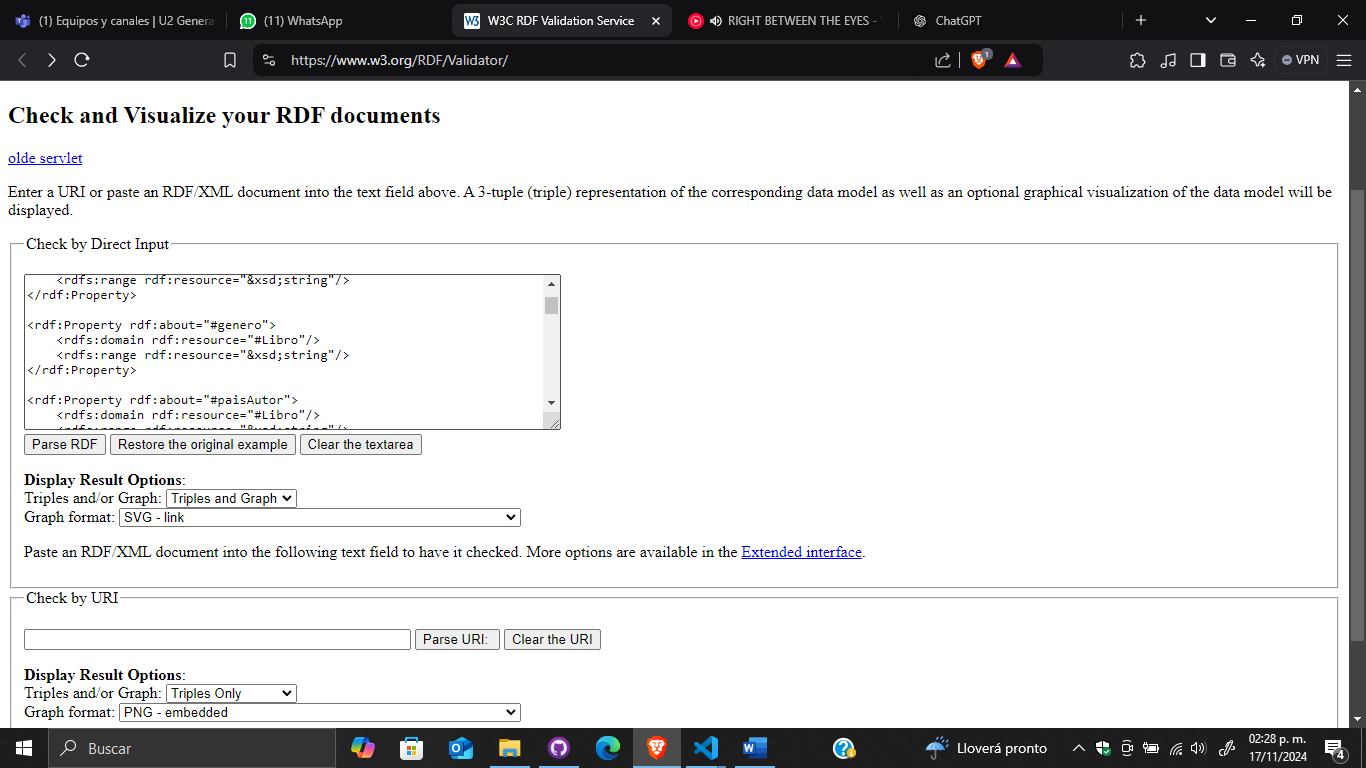
La cual sigue la estructura necesaria para generar correctamente el código RDF del préstamo (Fig. 1.8)



Este seria solamente un pequeño ejemplo de cómo podríamos facilitar en análisis de los datos de una manera mas clara, generando código semántico de manera automática a partir de una representación mas simple para el usuario, al igual que la gramática del usuario y del libro, la del préstamo representa mas claramente como funcionaria en un ámbito donde ambas están relacionadas.

# RDF Validator

En este caso después del análisis completo de como se generaron nuestros códigos RDF, podríamos hacer un análisis en una herramienta como RDF Validator, pero primero tenemos que saber que es, podemos decir que RDF Validator es una herramienta en línea para validar y analizar datos RDF, entonces haremos una pequeña prueba con nuestros datos generados (fig 1.9)



Bajo el contexto anterior podemos decir que la estructura del código es correcta, ya que la herramienta se encarga de validar las estructuras y que tenga sentido, por ejemplo, al hacer la validación nos muestra tres partes (Triples) del código: sujeto, predicado y objeto, los cuales juntos, forman una declaración sobre un recurso (Fig 1.10).



Después de esta forma podemos ver otra manera de expresar nuestro código RDF, como un “Gráfico de Modelo de Datos”, el cual es una representación visual del modelo RDF, donde los triples se ilustran como nodos y aristas en un grafo dirigido.

El modelo debido a su tamaño puede verse en el siguiente enlace:

[***https://www.w3.org/RDF/Validator/ARPServlet.tmp/servlet\_12699928583722703892.svg***](https://www.w3.org/RDF/Validator/ARPServlet.tmp/servlet_12699928583722703892.svg)

En lo anterior podemos ver como los ***nodos*** son la representación de los sujetos y objetos de los triples, además de también ver las ***aristas***, las cuales representan los predicados (propiedades o relaciones) que conectan los nodos.

Con lo anterior definido y probando, podemos llegar a la conclusión de que validar datos RDF, se garantiza que estos puedan ser interpretados y utilizados por diversas plataformas y agentes, mejorando la integración y el intercambio de información en un entorno digital, además de que esta validación fomenta prácticas de desarrollo más robustas y confiables, minimizando errores y mejorando la calidad de los sistemas basados en RDF.

# Integración en IDE

Para mayor practicidad se integro en un .jar la