19CS2107 ENTERPRISE PROGRAMMING

LAB-1

ID NO:190030350

Name:-D.Prem Venkat Pavan Kumar

Pre LabTask:

1)What is the full form of XML?

Ans: Extensible Markup Language.

2) What if the full form of DTD?

Ans:Document type Definition.

3) What is the full form of XSD?

Ans:XML Schema Definition.

4) What are the rules to be followed to create Well Formed XML Documents?

Ans: a) Prolog must be there (<?xml version="1.0" encoding="UTF-8"?>)

b) There must be only one root element.

c) Every tag must be properly closed.

d) Attributes values must be enclosed in ‘ ’ or “ ”.

5) Write down the functionalities and syntax of the below mentioned.

a)XML Naming Rules

XML elements must follow these naming rules:

Element names are case-sensitive

Element names must start with a letter or underscore

Element names cannot start with the letters xml (or XML, or Xml, etc)

Element names can contain letters, digits, hyphens, underscores, and periods

b)XML Element

An XML element is everything from (including) the element's start tag to (including) the element's end tag.

There can also be empty XML elements.
`Syntax:-<element-name> content </element-name>`
<registration_number>190030350</registration_number>

c)XML attributes

XML elements can have attributes, just like HTML.

Attributes are designed to contain data related to a specific element.

`Syntax:-<element-name attribute-name=”value”> content </element-name>`

<studentid=”190030350”>D P V PAVAN KUMAR</stud ent>

Value	Explanation

d) <! DOCTYPE >

- An XML document with correct syntax is called "Well Formed".
- An XML document validated against a DTD is both "Well Formed" and "Valid".
- **<!DOCTYPE note SYSTEM "note.dtd">**
- The DOCTYPE declaration above contains a reference to a DTD file.

e)<! ELEMENT >

- In a DTD, elements are declared with an ELEMENT declaration.
- In a DTD, XML elements are declared with the following syntax:

<!ELEMENT element-name category>

Or

<!ELEMENT element-name (element-content)>

- In a DTD, elements can also be declared as empty elements.
- **Syntax:- <!ELEMENT element-name EMPTY>**

f) <! ATTLIST >

- In a DTD, attributes are declared with an ATTLIST declaration.
- An attribute declaration has the following syntax:

<!ATTLIST element-name attribute-name attribute-type attribute-value> .

- There are different types of attributes.
- The **attribute-value** can be one of the following:

value	The default value of the attribute
#REQUIRED	The attribute is required
#IMPLIED	The attribute is optional
#FIXED value	The attribute value is fixed

g) simpleType

- The simpleType element defines a simple type and specifies the constraints and information about
  the values of attributes or text-only elements.
- The **parent elements** are **attributes, elements, lists, restriction, schema, union.**
- **Syntax: -**

```
<simpleType
 id=ID
 name=NCName
  any attributes
>

  (annotation?,(restriction|list|union))

</simpleType>
```

h) complexType

The complexType element defines a complex type. A complex type element is an XML element that contains other elements and/or attributes.

The **parent elements** are **elements, redefine, schema.**

Syntax: -

```
<complexType
   id=ID
  name=NCName
  abstract=true|false
  mixed=true|false
   block=(#all|list_of(extension|restriction))
   final=(#all|list_of(extension|restriction))
```

any_attributes
>(annotation?,(simpleContent|complexContent|((group|all| choice|sequence)?,((attribute|attributeGroup)*,anyAttribute?))))

</complexType>

4). Write down the functionalities of the below mentioned in DTD - Attributes
a)CDATA

- CDATA means character data.
- CDATA is text that will NOT be parsed by a parser. Tags inside the text will NOT be treated as markup and entities will not be expanded.

b.PCDATA

- PCDATA means parsed character data.
- Think of character data as the text found between the start tag and the end tag of an XML element.
- PCDATA is text that WILL be parsed by a parser. The text will be examined by the parser for entities and markup.
- Tags inside the text will be treated as markup and entities will be expanded.
- However, parsed character data should not contain any &, <, or > characters; these need to be represented by the & < and > entities, respectively.

c.Default value

Example:

DTD:

<!ELEMENT square EMPTY>

<!ATTLIST square width CDATA "0">

Valid XML: <square width="100" />

In the example above, the "square" element is defined to be an empty element with a "width" attribute of type CDATA. If no width is specified, it has a default value of 0.

d.#REQUIRED

Syntax:

<!ATTLIST element-name attribute-name attribute-type #REQUIRED>

Example:

DTD:

<!ATTLIST person number CDATA #REQUIRED>

Valid XML:

<person number="5677" />

Invalid XML:

<person />

- Use the #REQUIRED keyword if you don't have an option for a default value, but still want to force the attribute to be present.

e.#IMPLIED

- Syntax:

<!ATTLIST element-name attribute-name attribute-type #IMPLIED>

- DTD:

<!ATTLIST contact fax CDATA #IMPLIED>

Valid XML:

<contact fax="555-667788" />

Valid XML:

<contact />

•Use the #IMPLIED keyword if you don't want to force the author to include an attribute, and you don't have an option for a default value.

f.#FIXED value

•Synatx:

<!ATTLIST element-name attribute-name attribute-type #FIXED "value">

•DTD:

<!ATTLIST sender company CDATA #FIXED "Microsoft">

Valid XML:

<sender company="Microsoft" />

Invalid XML:

<sender company="XYZ" />

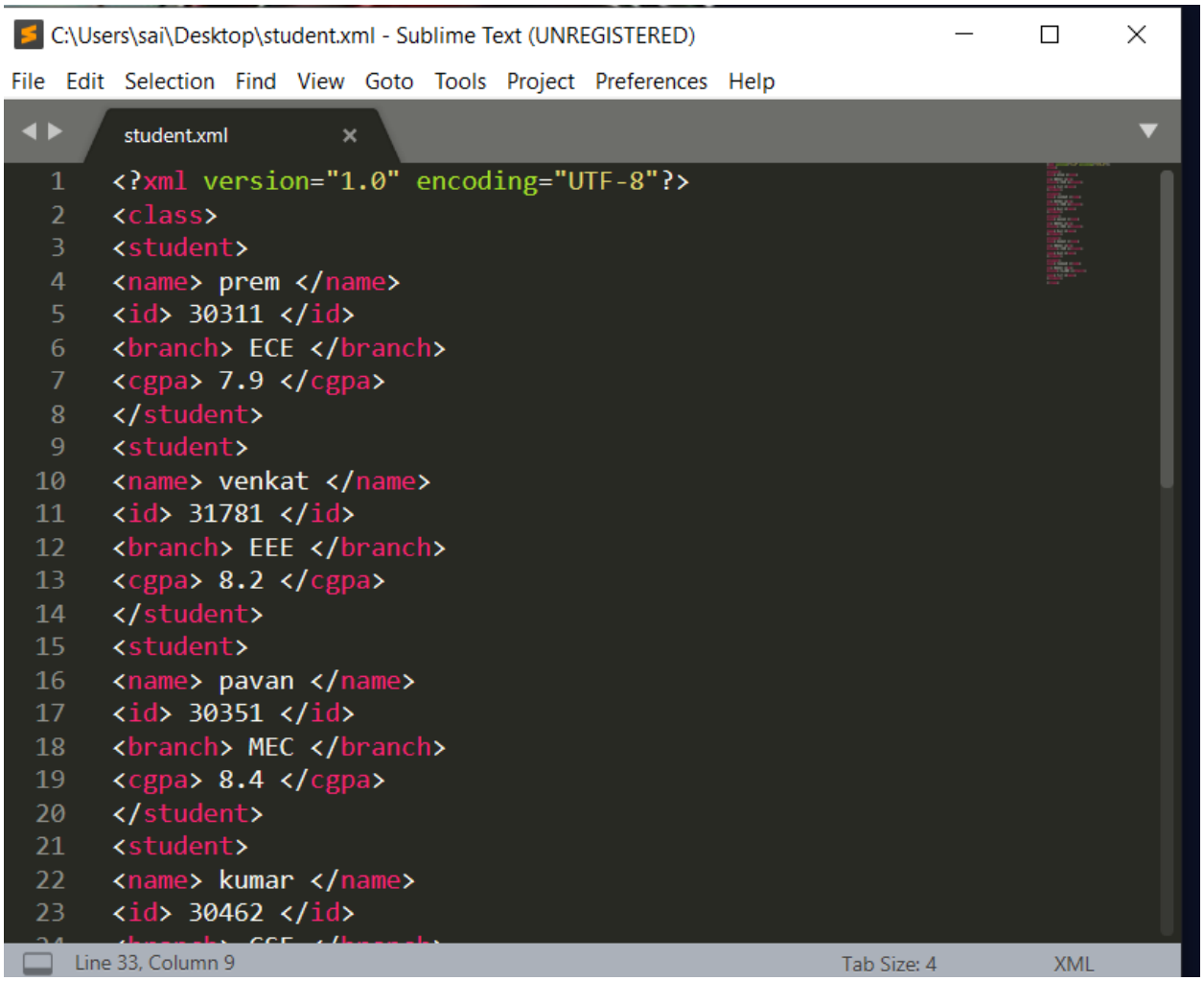
•Use the #FIXED keyword when you want an attribute to have a fixed value without allowing the author to change it. If an author includes another value, the XML parser will return an error.

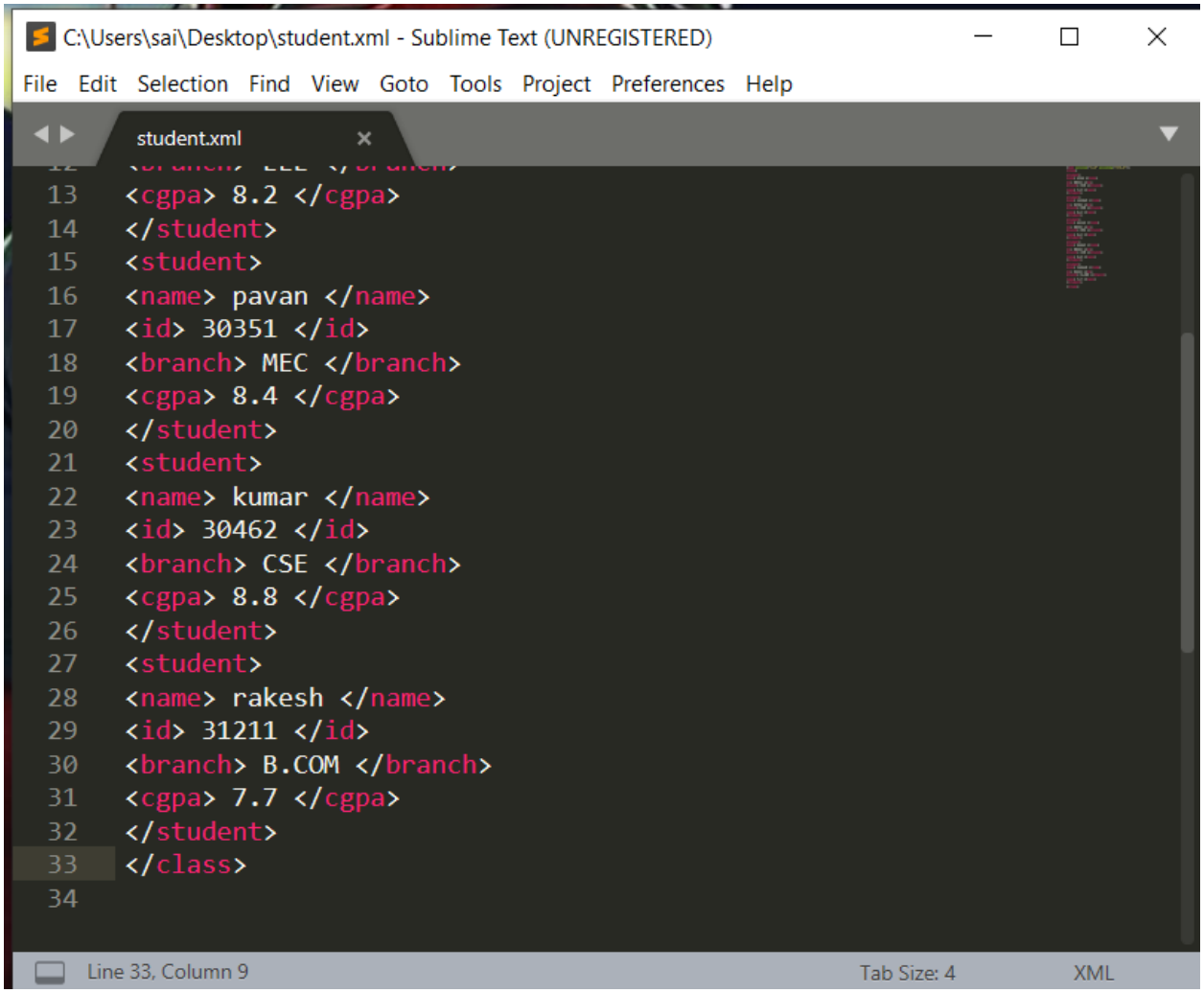
5)As you're in the very beginning stage of learning how to create a XML document. Write an XML to accept student details [Name, ID, Branch and CGPA] for minimum 5 students.

```
<class>
<student>
<name> ABC </name>
<id> 001 </id>
```

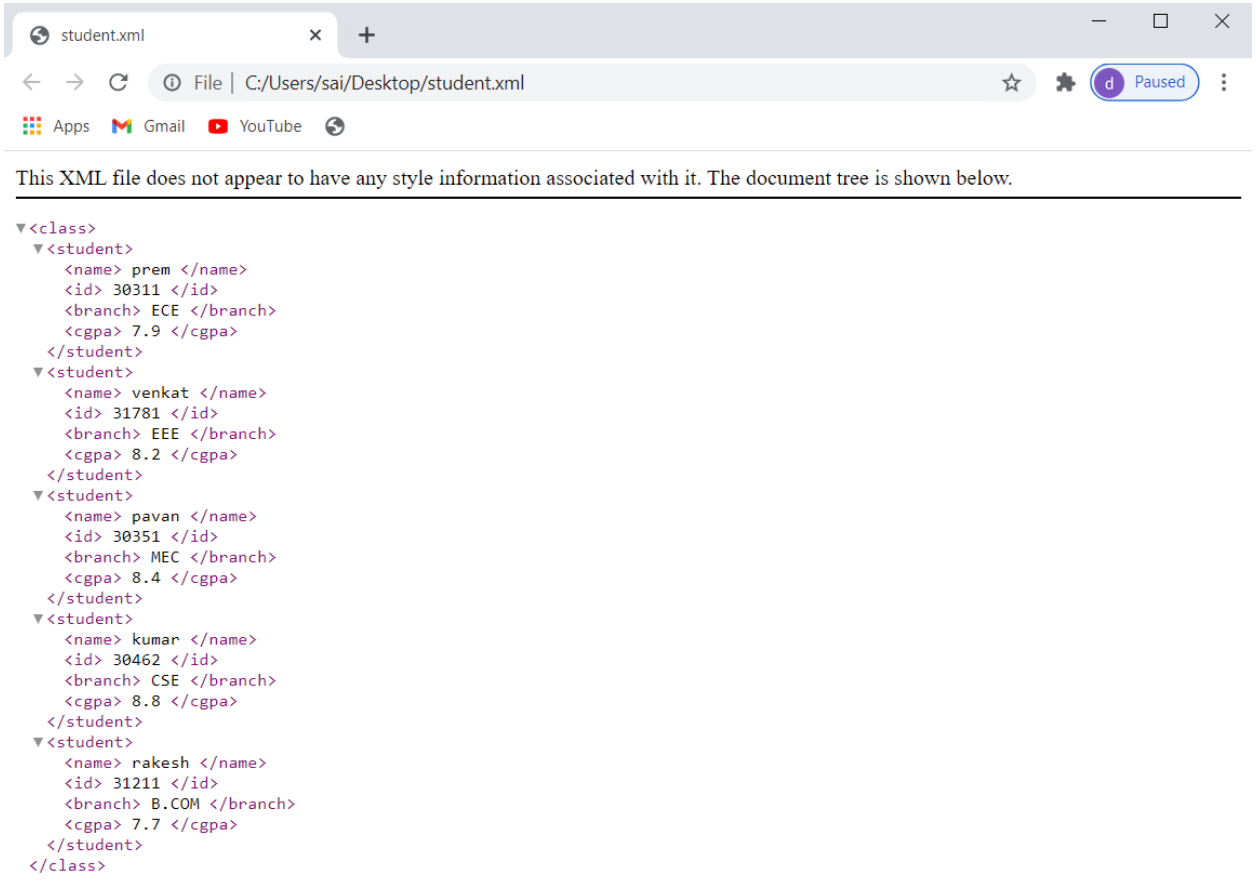
<branch> ECE </branch>
<cgpa> 97</cgpa>
</student>
<student>

Code:




C:\Users\sai\Desktop\student.xml - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
student.xml
<cgpa> 8.2 </cgpa>
</student>
<student>
<name> pavan </name>
<id> 30351 </id>
<branch> MEC </branch>
<cgpa> 8.4 </cgpa>
</student>
<student>
<name> kumar </name>
<id> 30462 </id>
<branch> CSE </branch>
<cgpa> 8.8 </cgpa>
</student>
<student>
<name> rakesh </name>
<id> 31211 </id>
<branch> B.COM </branch>
<cgpa> 7.7 </cgpa>
</student>
</class>
Line 33, Column 9
Tab Size: 4
XML

Output:


student.xml
File | C:/Users/sai/Desktop/student.xml
Paused
Apps
Gmail
YouTube
This XML file does not appear to have any style information associated with it. The document tree is shown below.
<class>
<student>
<name> prem </name>
<id> 30311 </id>
<branch> ECE </branch>
<cgpa> 7.9 </cgpa>
</student>
<student>
<name> venkat </name>
<id> 31781 </id>
<branch> EEE </branch>
<cgpa> 8.2 </cgpa>
</student>
<student>
<name> pavan </name>
<id> 30351 </id>
<branch> MEC </branch>
<cgpa> 8.4 </cgpa>
</student>
<student>
<name> kumar </name>
<id> 30462 </id>
<branch> CSE </branch>
<cgpa> 8.8 </cgpa>
</student>
<student>
<name> rakesh </name>
<id> 31211 </id>
<branch> B.COM </branch>
<cgpa> 7.7 </cgpa>
</student>
</class>

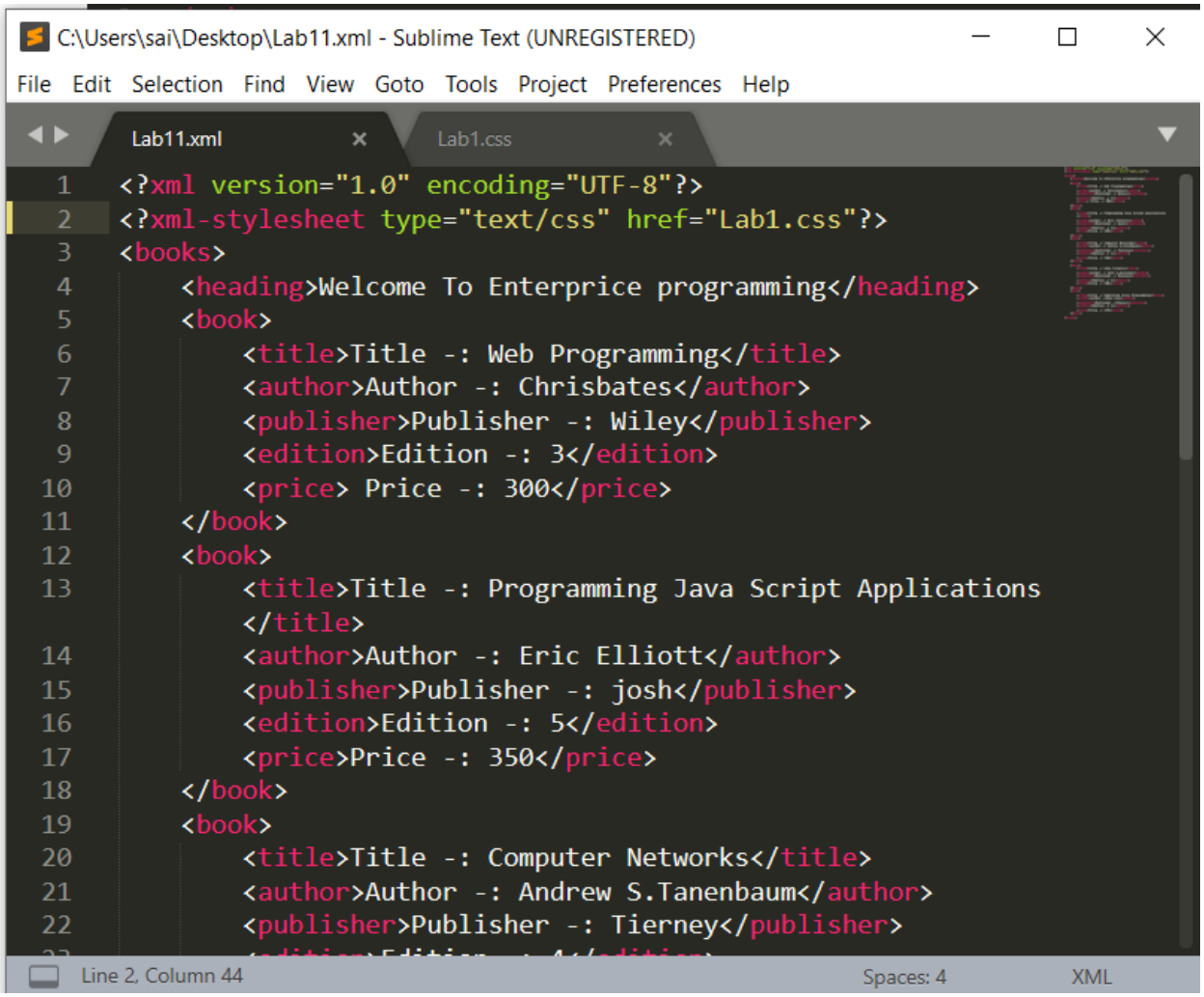
In Lab Task: -

a. Write a program for Books store, and the XML file is created that contains the information about five books and displaying the XML file using CSS.

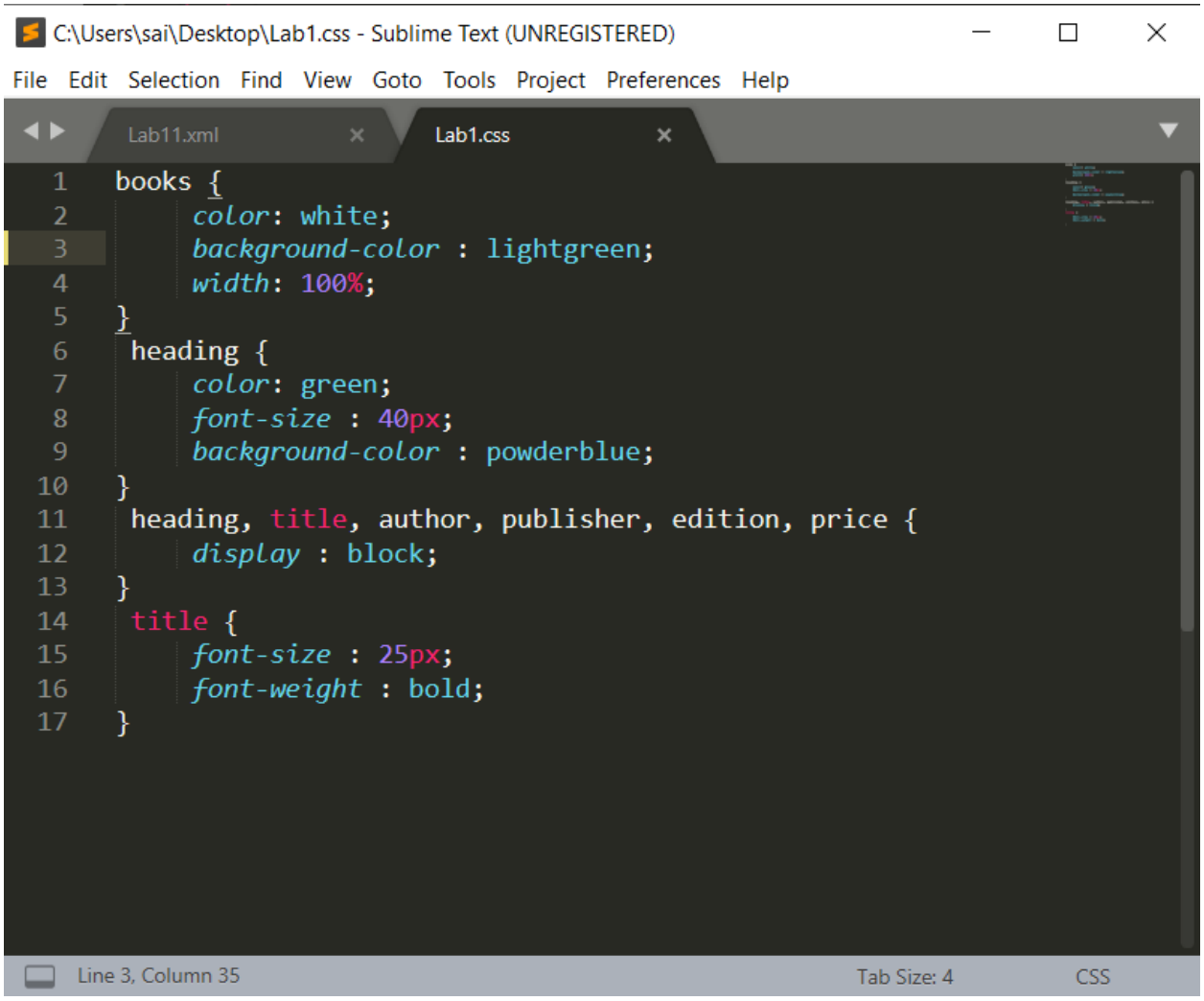
```
<books>
  <heading>Welcome To Enterprise Programming </heading>
  <book>
    <title>Title -: Web Programming</title>
    <author>Author -: Chrisbates</author>
    <publisher>Publisher -: Wiley</publisher>
    <edition>Edition -: 3</edition>
    <price> Price -: 300</price>
  </book>
….
….

<books>
```

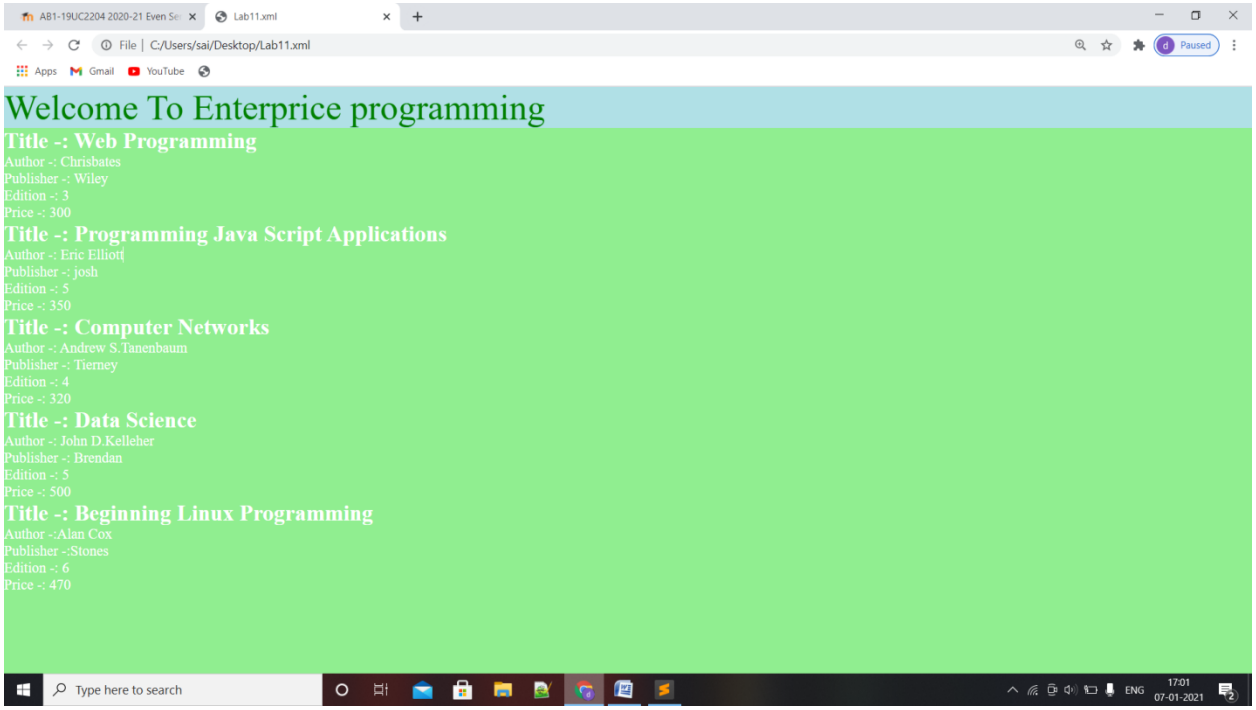
Code:


C:\Users\sai\Desktop\Lab11.xml - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Lab11.xml
Lab1.css
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/css" href="Lab1.css"?>
<books>
<heading>Welcome To Enterprice programming</heading>
<book>
<title>Title -: Web Programming</title>
<author>Author -: Chrisbates</author>
<publisher>Publisher -: Wiley</publisher>
<edition>Edition -: 3</edition>
<price> Price -: 300</price>
</book>
<book>
<title>Title -: Programming Java Script Applications
</title>
<author>Author -: Eric Elliott</author>
<publisher>Publisher -: josh</publisher>
<edition>Edition -: 5</edition>
<price>Price -: 350</price>
</book>
<book>
<title>Title -: Computer Networks</title>
<author>Author -: Andrew S.Tanenbaum</author>
<publisher>Publisher -: Tierney</publisher>
Line 2, Column 44
Spaces: 4
XML

Css:


C:\Users\sai\Desktop\Lab1.css - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
Lab11.xml
Lab1.css
books {
color: white;
background-color : lightgreen;
width: 100%;
}
heading {
color: green;
font-size : 40px;
background-color : powderblue;
}
heading, title, author, publisher, edition, price {
display : block;
}
title {
font-size : 25px;
font-weight : bold;
}
Line 3, Column 35
Tab Size: 4
CSS

Output:


AB1-19UC2204 2020-21 Even Se
Lab11.xml
File | C:/Users/sai/Desktop/Lab11.xml
Paused
Apps
Gmail
YouTube
Welcome To Enterprice programming
Title -: Web Programming
Author -: Chrisbates
Publisher -: Wiley
Edition -: 3
Price -: 300
Title -: Programming Java Script Applications
Author -: Eric Elliott
Publisher -: josh
Edition -: 5
Price -: 350
Title -: Computer Networks
Author -: Andrew S.Tanenbaum
Publisher -: Tierney
Edition -: 4
Price -: 320
Title -: Data Science
Author -: John D.Kelleher
Publisher -: Brendan
Edition -: 5
Price -: 500
Title -: Beginning Linux Programming
Author -:Alan Cox
Publisher -:Stones
Edition -: 6
Price -: 470
Type here to search
ENG
17:01
07-01-2021

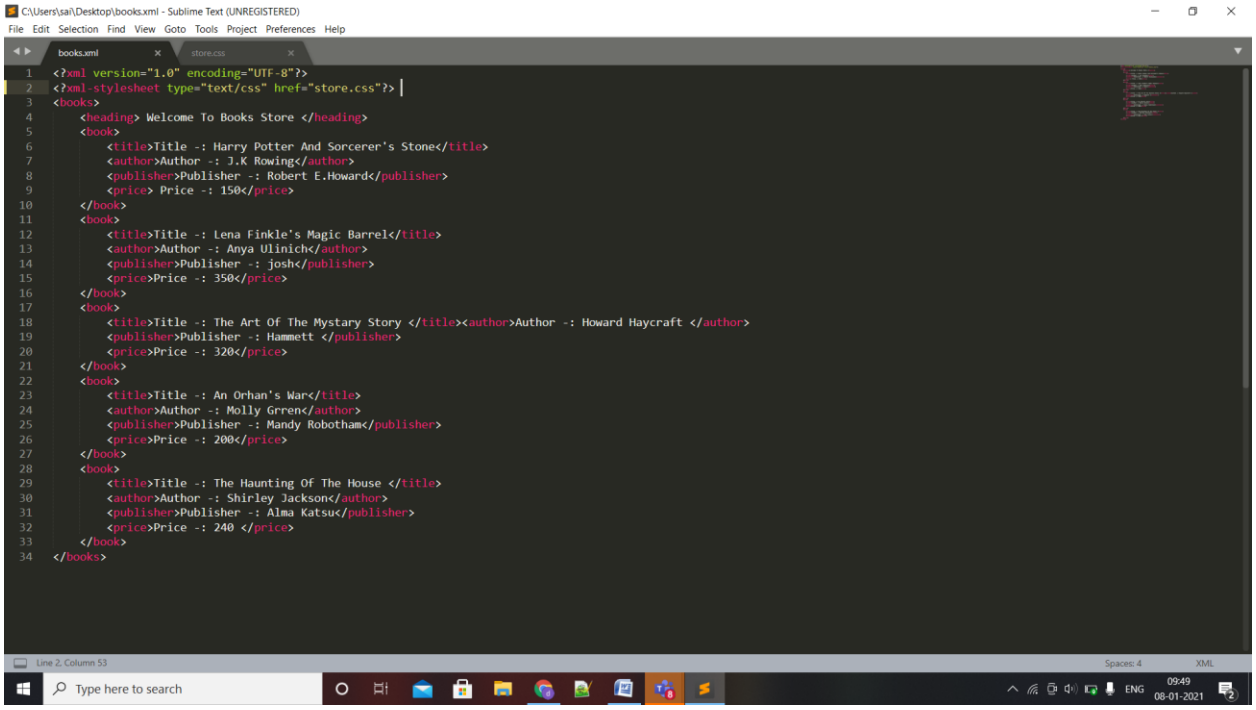
Post Lab Task:-

Write a program for Books store, and the XML file is created that contains the information about five books of different categories and displaying the XML file using CSS.

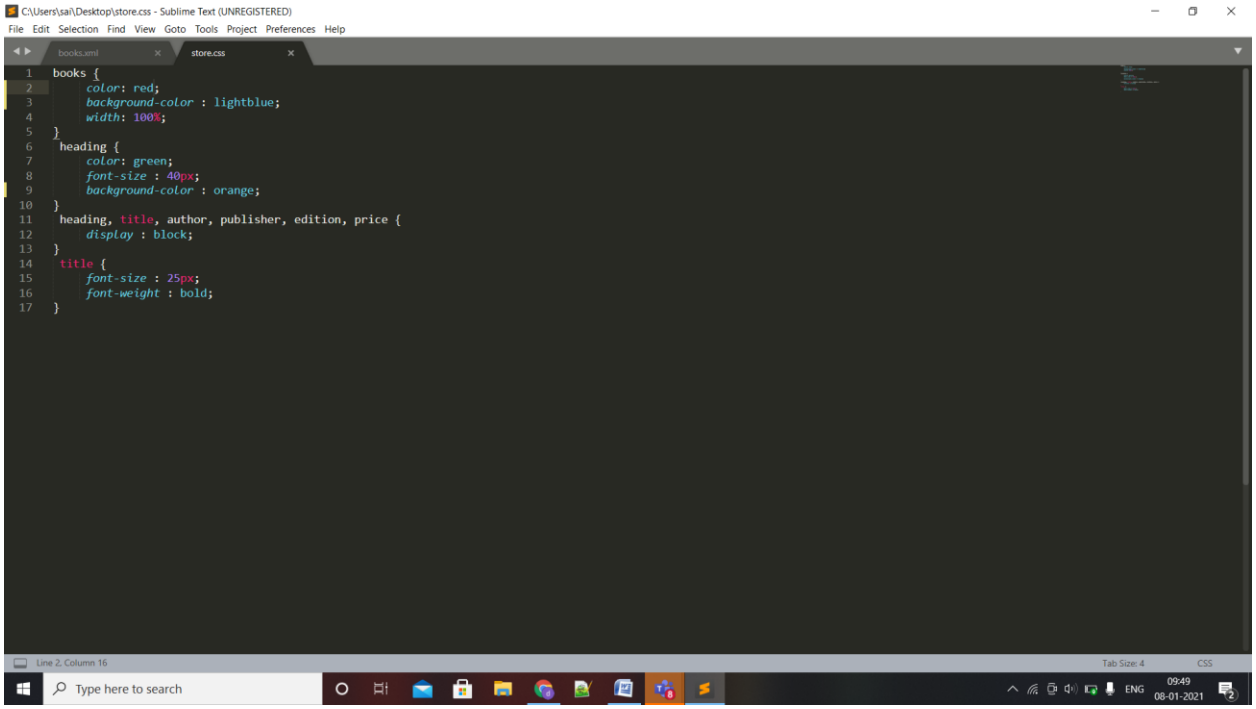
- Action and **Adventure**.
- Classics.
- Comic Book or **Graphic Novel**.
- Detective and Mystery.
- Fantasy.
- Historical Fiction.
- Horror.
- Literary Fiction.

Code:-

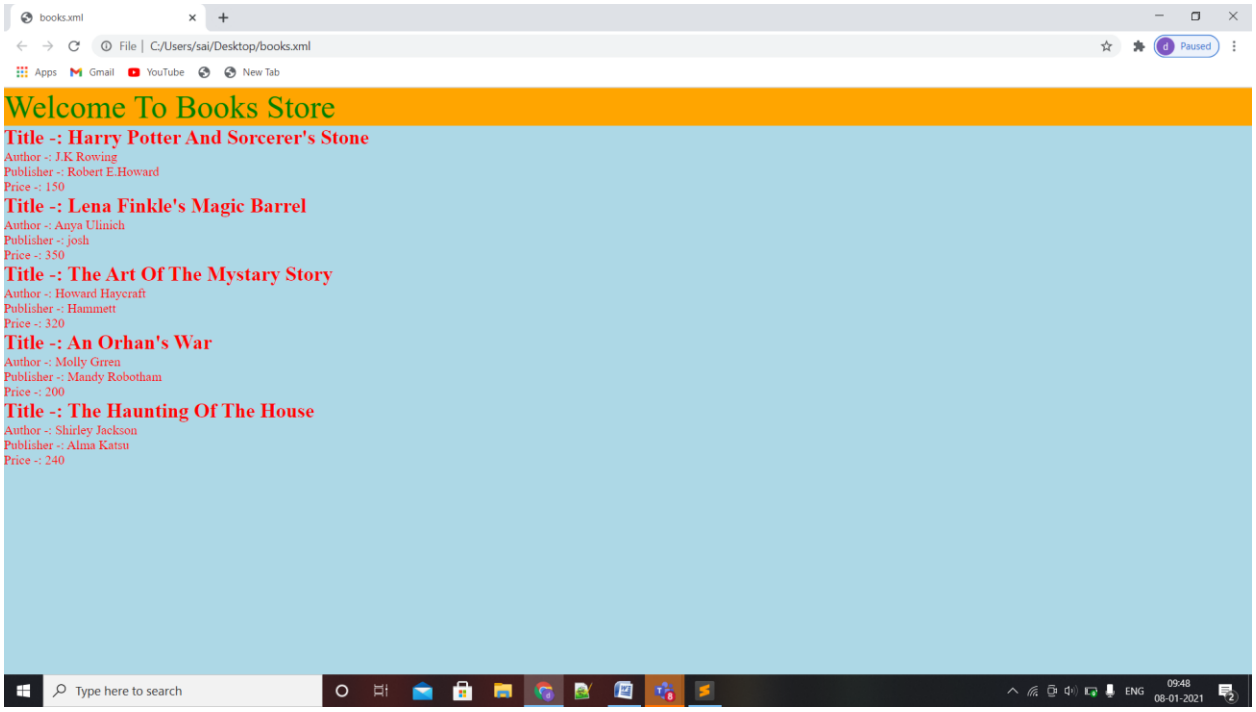
XML:-



Css:


C:\Users\sai\Desktop\store.css - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
books.xml
store.css
books {
color: red;
background-color : lightblue;
width: 100%;
}
heading {
color: green;
font-size : 40px;
background-color : orange;
}
heading, title, author, publisher, edition, price {
display : block;
}
title {
font-size : 25px;
font-weight : bold;
}
Line 2, Column 16
Tab Size: 4
CSS
Type here to search
ENG
09:49
08-01-2021

OUT PUT:-


books.xml
File | C:/Users/sai/Desktop/books.xml
Apps
Gmail
YouTube
New Tab
Paused
Welcome To Books Store
Title -: Harry Potter And Sorcerer's Stone
Author -: J.K Rowing
Publisher -: Robert E.Howard
Price -: 150
Title -: Lena Finkle's Magic Barrel
Author -: Anya Ulinich
Publisher -: josh
Price -: 350
Title -: The Art Of The Mystary Story
Author -: Howard Haycraft
Publisher -: Hammett
Price -: 320
Title -: An Orhan's War
Author -: Molly Grren
Publisher -: Mandy Robotham
Price -: 200
Title -: The Haunting Of The House
Author -: Shirley Jackson
Publisher -: Alma Katsu
Price -: 240
Type here to search
ENG
09:48
08-01-2021