

```

public void startElement(String uri, String localName, String qName, Attributes atts) { @Samuhdez17
    switch (qName) {
        case "nombre" -> tipoElemento = 'n';
        case "autor" -> tipoElemento = 'a';
        case "precio" -> tipoElemento = 'p';
        default -> tipoElemento = ' ';
    }
}

```

```

public void endDocument() { @Samuhdez17
    double precioFinal = 0.0;
    for (Double precio : precios) precioFinal += precio;

    System.out.println("\t+-----+-----+-----+");
    System.out.println("\t|          LIBRO          |          AUTOR          |    PRECIO    |");
    System.out.println("\t+-----+-----+-----+");

    for (int i = 0; i < nombres.size(); i++) {
        System.out.printf("\t| %-29s | %-29s | %10.2f |\n", nombres.get(i), autores.get(i), precios.get(i));
        System.out.println("\t+-----+-----+-----+");

        if (i == nombres.size() - 1) {
            System.out.printf("\t| %-59s | %10.2f |\n", "Precio total del pedido:", precioFinal);
            System.out.println("\t+-----+-----+-----+");
        }
    }
}

```

```

public void characters(char[] ch, int start, int length) throws SAXException { @Samuhdez17
    String contenido = new String(ch, start, length);
    contenido = contenido.replaceAll( regex: "[\t\n]", replacement: "");

    if (contenido.isBlank()) return;

    switch (tipoElemento) {
        case 'n' -> nombres.add(contenido);
        case 'a' -> autores.add(contenido);
        case 'p' -> {
            double precioConIva = Double.parseDouble(contenido) * 1.04;
            precios.add(precioConIva);
        }
    }
}

```