

<input type="checkbox"/> SSD41UE Haslinger	Name <u>Hillebrand Felix</u>	Aufwand in h <u>12</u>
<input checked="" type="checkbox"/> SSD42UE Traxler	Punkte _____	Kurzzeichen Tutor _____

1. XQuery (9 Punkte)

Übersetzen Sie die nachfolgenden textuellen Angaben in entsprechende XQuery-Ausdrücke, sodass die bei jeder Aufgabenstellung angegebene Ausgabe erzeugt wird. Testen Sie diese Ausdrücke auf Basis des bereitgestellten XML-Dokuments `ue3_1_fitnessdokument.xml`. Die Verwendung von XQuery 1.0 ist ausreichend. Der betroffene Namensraum aus dem XML-Dokument soll jeweils beibehalten werden.

- 1.1 Selektieren Sie die Messwert-Elemente, die zu Gewicht-Messungen gehören und sortieren Sie aufsteigend nach dem Datum der Messung. Umgeben Sie das gesamte Resultat mit einem Element `<M>`. (1,5 Punkt)

```
<M xmlns="http://www.fitness.at/fitness">
  <Messwert typ="Gewicht" wert="80.7" einheit="kg"/>
  <Messwert typ="Gewicht" wert="80.6" einheit="kg"/>
</M>
```

- 1.2 Selektieren Sie Messung-Elemente mit Blutdruck-Werten und verarbeiten Sie nur Messungen, die im Kommentar (Notiz) „maschinell“ (case-insensitiv) enthalten. Erstellen Sie ein neues Blutdruck-Element mit dem Zeitpunkt der Messung „zeit“ als Attribut. Fügen Sie zwei Kind-Elemente hinzu, „Sys“ für den systolischen Wert und „Dia“ für den diastolischen Wert. Packen Sie die Ausgabe in ein „RR“-Element. Verwenden Sie zur Formatierung der Ausgabe Element- und Attributkonstruktoren. (1,5 Punkte)

```
<RR xmlns="http://www.fitness.at/fitness">
  <Blutdruck zeit="2024-01-28T07:55:42">
    <Sys>121.0</Sys>
    <Dia>70.0</Dia>
  </Blutdruck>
  <Blutdruck zeit="2024-01-27T07:48:28">
    <Sys>115.0</Sys>
    <Dia>67.0</Dia>
  </Blutdruck>
</RR>
```

- 1.3 Erstellen Sie eine Aktivitätsübersicht für alle Geräte, mit denen bereits mindestens 2 Messvorgänge durchgeführt wurden, führen Sie dazu auch die Zeitpunkte an. Erstellen Sie hierfür ein Element „Aktivität“ und listen Sie darin „Gerät“-Elemente (mit der „id“ als Attribut) auf. Die Kindelemente „Online“ von „Gerät“ enthalten die Messzeitpunkte. (1,5 Punkte)

```
<Aktivität xmlns="http://www.fitness.at/fitness">
  <Gerät id="mg_bd1">
    <Online>2024-01-28T07:55:42</Online>
    <Online>2024-01-27T07:48:28</Online>
  </Gerät>
  <Gerät id="mg_wg1">
    <Online>2024-01-28T08:45:29</Online>
  </Gerät>
</Aktivität>
```



UNIVERSITY
OF APPLIED SCIENCES
UPPER AUSTRIA

SSD4 - UE03

Felix Hillebrand
97hilfel@gmail.com

Inhaltsverzeichnis

1	XQuery	1
2	XSLT	8
3	Legal	17

1 XQuery

Übersetzen Sie die nachfolgenden textuellen Angaben in entsprechende XQuery-Ausdrücke, sodass die bei jeder Aufgabenstellung angegebene Ausgabe erzeugt wird. Testen Sie diese Ausdrücke auf Basis des bereitgestellten XML-Dokuments `ue3_1_fitnessdokument.xml`. Die Verwendung von XQuery 1.0 ist ausreichend. Der betroffene Namensraum aus dem XML-Dokument soll jeweils beibehalten werden.

1. Selektieren Sie die Messwert-Elemente, die zu Gewicht-Messungen gehören und sortieren Sie aufsteigend nach dem Datum der Messung. Umgeben Sie das gesamte Resultat mit einem Element `<M>`. (1,5 Punkt)

```
1 <M xmlns="http://www.fitness.at/fitness">
2   <Messwert typ="Gewicht" wert="80.7" einheit="kg"/>
3   <Messwert typ="Gewicht" wert="80.6" einheit="kg"/>
4 </M>
```

Lösung:

```
1 (: Aufgabe 1.1 :)
2 declare default element namespace "http://www.fitness.at/fitness";
3
4 <M>
5   {
6     let $doc := doc("ue3_1_fitnessdokument.xml")
7     let $mss := $doc//Fitnessdokument//Vitaldaten//Messung//Messwert
8     for $mw in $mss
9       where $mw//@typ = "Gewicht"
10      return element Messwert {
11        attribute typ { $mw/@typ },
12        attribute wert { $mw/@wert },
13        attribute einheit { $mw/@einheit }
14      }
15   }
16 </M>
```

2. Selektieren Sie Messung-Elemente mit Blutdruck-Werten und verarbeiten Sie nur Messungen, die im Kommentar (Notiz) 'maschinell' (case-insensitiv) enthalten. Erstellen Sie ein neues Blutdruck-Element mit dem Zeitpunkt der Messung „zeit“ als Attribut. Fügen Sie zwei Kind-Elemente hinzu, „Sys“ für den systolischen Wert und „Dia“ für den diastolischen Wert. Packen Sie die Ausgabe in ein „RR“-Element. Verwenden Sie zur Formatierung der Ausgabe Element- und Attributkonstruktoren. (1,5 Punkte)

```

1 <RR xmlns="http://www.fitness.at/fitness">
2   <Blutdruck zeit="2024-01-28T07:55:42">
3     <Sys>121.0</Sys>
4     <Dia>70.0</Dia>
5   </Blutdruck>
6   <Blutdruck zeit="2024-01-27T07:48:28">
7     <Sys>115.0</Sys>
8     <Dia>67.0</Dia>
9   </Blutdruck>
10 </RR>

```

Lösung:

```

1 (: Aufgabe 1.2 :)
2 declare default element namespace "http://www.fitness.at/fitness";
3
4 <RR xmlns="http://www.fitness.at/fitness">
5   {
6     let $doc := doc("ue3_1_fitnessdokument.xml")
7     let $mss := $doc//Fitnessdokument//Vitaldaten//Messung
8     for $ms in $mss
9       where lower-case($ms//Notiz) = "maschinell" and $ms//@typ = "BlutdruckPuls"
10
11     return element Blutdruck {
12       attribute zeit { $ms/@zeitpunkt },
13       element Sys {
14         data($ms//Messwert[@typ="Systolisch"]//@wert)
15       },
16       element Dia {
17         data($ms//Messwert[@typ="Diastolisch"]//@wert)
18       }
19     }
20   }
21 </RR>

```

3. Erstellen Sie eine Aktivitätsübersicht für alle Geräte, mit denen bereits mindestens 2 Messvorgänge durchgeführt wurden, führen Sie dazu auch die Zeitpunkte an. Erstellen Sie hierfür ein Element 'Aktivität' und listen Sie darin 'Gerät'-Elemente (mit der 'id' als Attribut) auf. Die Kindelemente 'Online' von 'Gerät' enthalten die Messzeitpunkte. (1,5 Punkte)

```

1 <Aktivität xmlns="http://www.fitness.at/fitness">
2   <Gerät id="mg_bd1">
3     <Online>2024-01-28T07:55:42</Online>
4     <Online>2024-01-27T07:48:28</Online>
5   </Gerät>
6   <Gerät id="mg_wg1">
7     <Online>2024-01-28T08:45:29</Online>
8     <Online>2024-01-22T09:05:51</Online>
9   </Gerät>
10  <Gerät id="mg_bz1">
11    <Online>2024-01-28T08:44:47</Online>
12    <Online>2024-01-27T07:49:13</Online>
13  </Gerät>
14 </Aktivität>

```

Lösung:

```

1 (: Aufgabe 1.3 :)
2
3 declare default element namespace "http://www.fitness.at/fitness";
4
5 <Aktivität>
6 {
7   let $doc := doc("ue3_1_fitnessdokument.xml")
8
9   let $devices := $doc//Messgeräte//Messgerät
10  for $device in $devices
11  let $deviceId := data($device//@id)
12  let $measurements := $doc//Fitnessdokument//Vitaldaten//Messung[@messgerätId =
13    ↪ $deviceId]
14  where count($measurements) >= 2
15
16  return <Gerät id="{ $deviceId }">{
17    for $measurement in $measurements
18    return <Online>{data($measurement//@zeitpunkt)}</Online>
19  }</Gerät>
20 }
21 </Aktivität>

```

4. Geben Sie die Titel (wenn vorhanden) und Vor- und Nachnamen der Person in einem Element 'Name' aus. Fügen Sie auch das Geburtsdatum als Element 'Geburtsdatum' hinzu und berechnen Sie auch das Alter (Attribut 'alter'). Umgeben Sie die Elemente mit einem Tag 'Person'. (2 Punkte)

Ergänzen und verwenden Sie für die Berechnung des Alters folgenden Code:

```

1 let $dob := (:Date of birth / Geburtsdatum einsetzen:)
2 let $birth := number(concat(substring($dob, 1, 4), substring($dob, 6, 2),
3 substring($dob, 9, 2)))
4 let $now := (:now/ jetzt angeben:)
5 let $current := number(concat(substring($now, 1, 4), substring($now, 6, 2),
6 substring($now, 9, 2)))
7 let $age := floor (($current - $birth) div 10000)

```

```

1 <Person xmlns="http://www.fitness.at/person">
2     <Name>Sandra Sportlich</Name>
3     <Geburtsdatum alter="52">1971-08-24</Geburtsdatum>
4 </Person>
5 <!-- Titel zum Testen in XML-Dokument einfügen -->
6 <Person xmlns="http://www.fitness.at/person">
7     <Name>Mag. Sandra Sportlich MSc</Name>
8     <Geburtsdatum alter="52">1971-08-24</Geburtsdatum>
9 </Person>

```

Lösung:

```

1 (: Aufgabe 1.4 :)
2
3 declare namespace p = "http://www.fitness.at/person";
4
5 let $doc := fn:doc("ue3_1_fitnessdokument.xml")
6 for $pers in $doc//p:Person
7
8     let $dob := $pers//@geburtsdatum
9     let $birth := number(concat(substring($dob, 1, 4), substring($dob, 6, 2),
10 substring($dob, 9, 2)))
11     let $now := string(current-dateTime())
12     let $current := number(concat(substring($now, 1, 4), substring($now, 6, 2),
13 substring($now, 9, 2)))
14     let $age := floor (($current - $birth) div 10000)
15
16     let $vorTitel := data($pers//p:Titel[@position = "vor"])
17     let $nachTitel := data($pers//p:Titel[@position = "nach"])
18
19     return <Person>
20         <Name>
21             {
22                 $vorTitel,
23                 data($pers//p:Vorname),
24                 data($pers//p:Nachname),

```

```
25         $nachTitel
26     }
27 </Name>
28 <Geburtsdatum age="{ $age }">
29     {
30         data($pers//@geburtsdatum)
31     }
32 </Geburtsdatum>
33 </Person>
```

5. Erstellen Sie ein XML-Fragment "Vitalwerte", das alle Messungen mit Messwerten listet. Ergänzen Sie die Messwerte um ein Attribut „flag“ wenn der Messwert außerhalb angegebener Grenzwerte liegt. Berücksichtigen Sie nur Grenzwerte für den passenden Mess-Typ und beachten Sie auch das Gültigkeitsdatum. Liegt kein zutreffender Grenzwert vor bzw. ist der gemessene Wert innerhalb der Grenzwerte, entfällt das Attribut "flag". Das Attribut enthält "L" für Werte unter dem unteren Grenzwert bzw. "H" für Werte über dem oberen Grenzwert. (2,5 Punkte)

Hinweis: Verwenden Sie `xs:number(...)` für die Umwandlung der Grenzwerte zum Wertevergleich.

```

1 <Vitalwerte xmlns="http://www.fitness.at/fitness">
2   <Messung typ="BlutdruckPuls" zeitpunkt="2024-01-28T07:55:42">
3     <Messwert typ="Systolisch" wert="121.0" einheit="mmHg"/>
4     <Messwert typ="Diastolisch" wert="70.0" einheit="mmHg"/>
5     <Messwert typ="Puls" wert="79.0" einheit="Schläge/min"/>
6   </Messung>
7   <Messung typ="Gewicht" zeitpunkt="2024-01-28T08:45:29">
8     <Messwert typ="Gewicht" wert="80.6" einheit="kg"/>
9   </Messung>
10  <Messung typ="Schritte" zeitpunkt="2024-01-28T00:00:00">
11    <Messwert typ="Schritte" wert="9800" einheit=""/>
12  </Messung>
13  <Messung typ="BlutdruckPuls" zeitpunkt="2024-01-27T07:48:28">
14    <Messwert typ="Systolisch" wert="115.0" einheit="mmHg"/>
15    <Messwert typ="Diastolisch" wert="67.0" einheit="mmHg"/>
16  </Messung>
17  <Messung typ="Gewicht" zeitpunkt="2024-01-22T09:05:51">
18    <Messwert typ="Gewicht" wert="80.7" einheit="kg"/>
19  </Messung>
20  <Messung typ="Schritte" zeitpunkt="2024-01-20T00:00:00">
21    <Messwert typ="Schritte" wert="6320" einheit="pro Tag" flag="L"/>
22  </Messung>
23  <Messung typ="Blutzucker" zeitpunkt="2024-01-28T08:44:47">
24    <Messwert typ="Blutzucker" wert="155.5" einheit="mg/dl"/>
25  </Messung>
26  <Messung typ="Blutzucker" zeitpunkt="2024-01-27T07:49:13">
27    <Messwert typ="Blutzucker" wert="201.5" einheit="mg/dl" flag="H"/>
28  </Messung>
29 </Vitalwerte>

```

Lösung:

```

1 (: Aufgabe 1.5 :)
2
3 declare default element namespace "http://www.fitness.at/fitness";
4
5 <Vitalwerte>{
6   let $doc := fn:doc("ue3_1_fitnessdokument.xml")
7   let $limits := $doc//Fitnessdokument//Grenzwertliste//Grenzwert

```



```
8  let $measurements := $doc//Fitnessdokument//Vitaldaten//Messung
9
10 for $measurement in $measurements
11 let $typ := data($measurement//@typ)
12
13 return element { node-name($measurement) } {
14     $measurement/.*,
15     if ($limits//@typ = $typ)
16     then if (xs:decimal(data($limits[@typ = $typ]//Schwellwert[@unten]//@wert)) <
17         ↪ xs:decimal(data($measurement//Messwert//@wert)))
18         then attribute flag { "L" }
19         else if (xs:decimal(data($limits[@typ = $typ]//Schwellwert[@oben]//@wert)) >
20             ↪ xs:decimal(data($measurement//Messwert//@wert)))
21             then attribute flag { "H" }
22             else ()
23         else (),
24     for $node in $measurement/node()
25     where not(local-name($node) = "Notiz")
26     return $node
27 }
28
29 }</Vitalwerte>
```

2 XSLT

Erstellen Sie ein Stylesheet (fitnessdokument.xslt), das zur Transformation des XML- Dokuments ue3_2_fitnessdokumen in ein HTML-Dokument verwendet werden kann. Ergänzen Sie hierfür die Vorgabe in fitnessdokument.xslt. Sie müssen hierbei keine CSS- Klassen explizit verwenden. Achten Sie darauf, dass das Stylesheet für verschiedene Fitnessdokumente funktioniert und keine sinnlosen Ausgaben produziert - z.B. 0-n Medikamente, 0-n akademische Titel, ...

Hinweis: Verwenden Sie XSLT 2.0, um auch auf fortgeschrittene Formatierungsfunktionen Zugriff zu erhalten (z.B. <https://www.w3.org/TR/xslt20/#format-date> für die Anpassung der Zeitstempel; wie z.B. `format-dateTime(..)` oder `format-date(datum, "[D].[M].[Y]")`)

- a. Binden Sie das Dokument templates.xslt in Ihr fitnessdokument.xslt ein, sodass die benannten Templates verwendet werden können und setzen Sie diese nach Möglichkeit ein.
- b. Geben Sie zu Beginn die Basisdaten des Fitnessdokuments, sowie die Personendaten aus.
- c. Erstellen Sie ein benanntes Template printMessungTabelle. Als Parameter des Templates definieren und verwenden Sie den Messungs-Typ und die Messungen. Verwenden Sie nach Möglichkeit bereits vorgegebene benannte Templates printUserFormattedDateTime, printWertEinheit und getSeriennummerForMessgerätId. Sortieren Sie die einzelnen Messungen.
- d. Wird der Knoten Vitaldaten selektiert, rufen Sie das neue Template printMessungTabelle auf.
- e. Ergänzen Sie Templates zur Ausgabe der Messgeräte bzw. des jeweiligen Messgeräts bzw. Gerätetyp, Hersteller, Modellnummer, Seriennummer.
- f. Ergänzen Sie die jeweiligen Templates, sodass die Medikationsliste bzw. die Medikation entsprechend der Vorgabe ausgegeben wird.
- g. Ergänzen Sie in templates.xslt die Definition für den unteren bzw. oberen Schwellwert ausgehend vom Parameter messung. Sie können davon ausgehen, dass der Parameter messung einer Messung entspricht.
- h. Ergänzen Sie weiters: Liegt der Wert ober bzw. unterhalb der Schwellwerte soll ++ bzw. - angezeigt werden.
- i. Verwenden Sie das Dokument ue3_2_Fitnessdokument_spezial.xml und prüfen Sie, ob Ihr Stylesheet aus Beispiel 1 auch Sonderfälle ordentlich verarbeiten kann. Passen Sie Ihr fitnessdokument.xslt oder templates.xslt gegebenenfalls diesbezüglich an.
- j. Die Verarbeitung mit XSLT kann sowohl pull- als auch push-basiert erfolgen. Ihr erstelltes Stylesheet verwendet eine Mischung aus diesen beiden Verarbeitungsarten. Kennzeichnen Sie die jeweiligen Templates bzw. Abschnitte in Ihrem Stylesheet mit der Art der Verarbeitung (push-basiert / pull-basiert).

Fitnessdokument:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsl:stylesheet version="2.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  ↪ xmlns:fo="http://www.w3.org/1999/XSL/Format"
  ↪ xmlns:xs="http://www.w3.org/2001/XMLSchema"
  ↪ xmlns:fn="http://www.w3.org/2005/xpath-functions"
  ↪ xpath-default-namespace="http://www.fitness.at">
```

```

3      <xsl:output method="html" encoding="UTF-8" indent="yes"/>
4      <!--1a Binden Sie das templates.xslt in dieses XSLT ein, sodass die
5      ↳ entsprechenden Templates verwendet werden können.-->
6      <xsl:include href="templates.xslt"/>
7
8      <xsl:template match="/">
9          <html>
10             <head>
11                 <meta charset="UTF-8"/>
12                 <title>Fitnessdokument</title>
13                 <style type="text/css">
14                     table.t2 {
15                         border-collapse: collapse;
16                     }
17                     .t2 caption {
18                         padding-bottom: 0.5em;
19                         font-weight: bold;
20                         font-size: 16px;
21                     }
22                     .t2 th, .t2 td {
23                         padding: 4px 8px;
24                         border: 2px solid #fff;
25                         background: #bdd2d9;
26                     }
27                     .t2 thead th {
28                         padding: 2px 8px;
29                         background: #328da8;
30                         text-align: left;
31                         font-weight: bold;
32                         color: #fff;
33                     }
34                     .t2 tr *:nth-child(3), .t2 tr *:nth-child(4) {
35                         text-align: right;
36                     }
37                 </style>
38             </head>
39             <body>
40                 <!--1b Geben Sie zu Beginn die Basisdaten des
41                 ↳ Fitnessdokuments, sowie die Personendaten
42                 ↳ aus.-->
43                 <xsl:apply-templates select="/Fitnessdokument"/>
44             </body>
45         </html>
46     </xsl:template>
47
48     <!--1b Geben Sie zu Beginn die Basisdaten des Fitnessdokuments, sowie die
49     ↳ Personendaten aus.-->
50     <xsl:template match="Fitnessdokument">

```

```

47         <h1>Fitnessdokument (Version <xsl:value-of select="@version" />) vom
         ↪ <xsl:call-template name="printUserFormattedDateTime"><xsl:with-param
         ↪ name="dateTimeToFormat" select="@erzeugt" /></xsl:call-template></h1>
48     <hr></hr>
49     <h2>Persönliche Daten</h2>
50     <br></br>
51     <xsl:apply-templates select="Person" />
52     <hr></hr>
53     <xsl:apply-templates select="Vitaldaten" />
54     <xsl:apply-templates select="Messgeräte" />
55     <hr></hr>
56     <xsl:apply-templates select="Medikationsliste" />
57 </xsl:template>
58
59 <xsl:template match="Person">
60 <p>
61     <xsl:value-of select="Titel[@position='vor']" />
62     <xsl:text> </xsl:text>
63     <xsl:value-of select="Vorname" />
64     <xsl:text> </xsl:text>
65     <xsl:value-of select="Nachname" />
66     <xsl:text>, </xsl:text>
67     <xsl:value-of select="Titel[@position='nach']" />
68 </p>
69
70 <p>
71     <b>Geschlecht:</b>
72     <xsl:value-of select="@geschlecht" />
73     <xsl:text> | </xsl:text>
74     <b>geboren am: </b>
75     <xsl:value-of select="format-date(@geburtsdatum,
76     ↪ ' [D01] . [M1] . [Y0001] ')"></xsl:value-of>
77     <xsl:text> | </xsl:text>
78     <b>SVNR:</b>
79     <xsl:value-of select="@svnr" />
80 </p>
81 </xsl:template>
82
83 <!--1c Erstellen Sie ein benanntes Template printMessungTabelle.-->
84 <xsl:template name="printMessungTabelle">
85     <xsl:param name="messungTyp" />
86     <xsl:param name="messungen" />
87
88     <table class="t2">
89         <thead>
90             <tr>
91                 <th>Zeitpunkt</th>

```

```

92         <th>Notiz/Messbedingung</th>
93         <th>Messgerät (SNr.)</th>
94     </tr>
95 </thead>
96
97 <tbody>
98     <xsl:for-each select="$messungen">
99         <xsl:sort select="@zeitpunkt"/>
100         <xsl:variable name="zeitpunkt" select="@zeitpunkt"/>
101         <xsl:variable name="wert" select="Messwert"/>
102         <xsl:variable name="notiz" select="Notiz | Messbedingung"/>
103         <xsl:variable name="messgerätId" select="@messgerätId"/>
104         <xsl:variable name="messgerät"
105             ↪ select="/Fitnessdokument/Messgeräte/Messgerät[@id=$messgerätId]"/>
106         <tr>
107             <td><xsl:call-template
108                 ↪ name="printUserFormattedDateTime"><xsl:with-param
109                 ↪ name="dateTimeToFormat"
110                 ↪ select="$zeitpunkt"/></xsl:call-template></td>
111             <td><xsl:call-template name="printWertEinheit"><xsl:with-param
112                 ↪ name="messwerte" select="$wert" /></xsl:call-template></td>
113             <td><xsl:value-of select="$notiz"/></td>
114             <td><xsl:value-of select="$messgerätId"/> (<xsl:value-of
115                 ↪ select="$messgerät/Seriennummer"/>)</td>
116         </tr>
117     </xsl:for-each>
118 </tbody>
119 </table>
120 </xsl:template>
121
122 <!--1d Wird der Knoten Vitaldaten selektiert, rufen Sie das neue Template
123 ↪ printMessungTabelle auf.-->
124 <xsl:template match="Vitaldaten">
125     <xsl:for-each-group select="./Messung" group-by="@typ">
126         <xsl:variable name="typ">
127             <xsl:choose>
128                 <xsl:when
129                     ↪ test="current-grouping-key()='BlutdruckPuls'">Blut
130                 <xsl:otherwise>
131                     <xsl:value-of
132                         ↪ select="current-grouping-key()"/>
133                 </xsl:otherwise>
134             </xsl:choose>
135         </xsl:variable>
136         <h2>
137             <xsl:value-of select="$typ"/>
138         </h2>

```

```

131     <xsl:call-template name="printMessungTabelle">
132         <xsl:with-param name="messungTyp" select="$typ" />
133         <xsl:with-param name="messungen" select="current-group()" />
134     </xsl:call-template>
135     </xsl:for-each-group>
136 </xsl:template>
137
138 <!--1e Ergänzen Sie Templates zur Ausgabe der Messgeräte bzw. des jeweiligen
139      ↳ Messgeräts bzw. Gerätetyp, Hersteller, Modellnummer, Seriennummer-->
140 <xsl:template match="Messgeräte">
141     <h2>Messgeräte</h2>
142
143     <table class="t2">
144         <thead>
145             <tr>
146                 <th>Gerätetyp</th>
147                 <th>Hersteller</th>
148                 <th>Modellnummer</th>
149                 <th>Seriennummer</th>
150             </tr>
151         </thead>
152
153         <tbody>
154             <xsl:for-each select="Messgerät">
155                 <tr>
156                     <th><xsl:value-of select="Gerätetyp" /></th>
157                     <th><xsl:value-of select="Hersteller" /></th>
158                     <th><xsl:value-of select="Modellnummer" /></th>
159                     <th><xsl:value-of select="Seriennummer" /></th>
160                 </tr>
161             </xsl:for-each>
162         </tbody>
163     </table>
164 </xsl:template>
165
166 <!--1f Ergänzen Sie die jeweiligen Templates, sodass die Medikationsliste bzw.
167      ↳ die Medikation entsprechend der Vorgabe ausgegeben wird-->
168 <xsl:template match="Medikationsliste">
169     <h2>Medikationsliste</h2>
170     <table class="t2">
171         <thead>
172             <tr>
173                 <th>Medikament</th>
174                 <th>Einnahme</th>
175                 <th>Dosierung</th>
176                 <th>Hinweis</th>
177                 <th>Zusatzinformation</th>
178             </tr>

```

```

177         </thead>
178         <tbody>
179             <xsl:apply-templates/>
180         </tbody>
181     </table>
182 </xsl:template>
183
184 <xsl:template match="Medikation">
185     <tr>
186         <td><b><xsl:value-of select="."/></b></td>
187         <td><xsl:value-of select="@einnahme"/></td>
188         <td><xsl:value-of select="@dosierung"/></td>
189         <td><xsl:value-of select="@hinweis"/></td>
190     <td>
191         <xsl:if test="@von">
192             <b>Start: </b><xsl:value-of select="@von"/>
193             <br/>
194         </xsl:if>
195         <xsl:if test="@bis">
196             <b>Ende: </b><xsl:value-of select="@bis"/>
197             <br/>
198         </xsl:if>
199         <b>Anwendung: </b><xsl:value-of select="@anwendung"/>
200     </td>
201 </tr>
202 </xsl:template>
203 </xsl:stylesheet>

```

Templates:

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsl:stylesheet version="2.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
   ↪ xmlns:fo="http://www.w3.org/1999/XSL/Format"
3 xmlns:xs="http://www.w3.org/2001/XMLSchema"
   ↪ xmlns:fn="http://www.w3.org/2005/xpath-functions"
4 xpath-default-namespace="http://www.fitness.at">
5     <!--1a Binden Sie dieses XSLT in fitnessdokument.xslt ein und verwenden Sie die
   ↪     entsprechenden Templates-->
6     <xsl:template name="printUserFormattedDateTime">
7         <xsl:param name="dateTimeToFormat"/>
8         <xsl:param name="dateToFormat"/>
9         <xsl:variable name="dateTimeFormat" select="'[D]. [MNn] [Y] um
   ↪     [H01]:[m01] Uhr'"/>
10        <xsl:variable name="dateFormat" select="'[D]. [M]. [Y] '"/>
11        <xsl:variable name="language" select="'de'"/>
12        <xsl:if test="exists($dateTimeToFormat) and ($dateTimeToFormat ne '')">
13            <xsl:value-of select="format-dateTime($dateTimeToFormat,
   ↪     $dateTimeFormat, $language, (), ())"/>

```

```

14         </xsl:if>
15         <xsl:if test="exists($dateToFormat) and ($dateToFormat ne '')">
16             <xsl:value-of select="format-date($dateToFormat, $dateFormat,
17                 ↪ $language, (), ())"/>
18         </xsl:if>
19     </xsl:template>
20
21     <xsl:template name="printWertEinheit">
22         <xsl:param name="messwerte"/>
23
24         <xsl:for-each select="$messwerte">
25             <xsl:variable name="messwert" select="."/>
26
27             <xsl:if test="$messwert/../../@typ = 'BlutdruckPuls'">
28                 <xsl:value-of select="concat($messwert/@typ, ': ')" />
29             </xsl:if>
30
31             <xsl:variable name="wert"
32                 ↪ select="fn:number($messwert/@wert)" />
33             <xsl:variable name="einheit" select="$messwert/@einheit" />
34
35             <!--1g Ergänzen Sie die Definition für den unteren bzw. oberen
36                 ↪ Schwellwert ausgehend vom Parameter messung. Sie können davon
37                 ↪ ausgehen, dass der Parameter messung einer Messung
38                 ↪ entspricht.-->
39             <xsl:variable name="messung" select=".." />
40             <xsl:variable name="messwertTyp" select="$messwert/@typ" />
41
42             <xsl:variable name="untererSchwellwert"
43                 ↪ select="number(/Fitnessdokument/Grenzwertliste/Grenzwert[@typ =
44                 ↪ $messwertTyp and @von <= $messung/@zeitpunkt and @bis >=
45                 ↪ $messung/@zeitpunkt]/Schwellwert[@typ = 'unten']/@wert)" />
46             <xsl:variable name="obererSchwellwert"
47                 ↪ select="number(/Fitnessdokument/Grenzwertliste/Grenzwert[@typ =
48                 ↪ $messwertTyp and @von <= $messung/@zeitpunkt and @bis >=
49                 ↪ $messung/@zeitpunkt]/Schwellwert[@typ = 'oben']/@wert)" />
50
51             <!--2. Sonderfälle: abprüfen, ob Wert plausibel-->
52             <xsl:if test="(exists($wert)) and ($wert >= 0)">
53                 <xsl:value-of select="$wert" />
54             </xsl:if>
55
56             <!--2. Sonderfälle: abprüfen, ob Einheit angegeben wird-->
57             <xsl:if test="(exists($einheit)) and ($einheit ne '')">
58                 (<xsl:value-of select="$einheit" />)
59             </xsl:if>

```



```
50          <!--1h Ergänzen Sie: Liegt der Wert ober bzw. unterhalb der
           ↳ Schwellwerte soll ++ bzw. -- angezeigt werden-->
51      <xsl:if test="$wert >= $obererSchwellwert">
52          ++
53      </xsl:if>
54      <xsl:if test="$wert <= $untererSchwellwert">
55          --
56      </xsl:if>
57
58          <br/>
59      </xsl:for-each>
60  </xsl:template>
61
62
63
64      <xsl:template name="getSeriennummerForMessgerätId">
65          <xsl:param name="messung"/>
66          <xsl:value-of
           ↳ select="$messung/../../Messgeräte/Messgerät[@id=$messung/@messgerätId]/Ser
67      </xsl:template>
68
69      <xsl:template name="Formatierung">
70          <hr/>
71      </xsl:template>
72
73
74  </xsl:stylesheet>
```

Screenshot:

Fitnessdokument (Version 3) vom 1. März 2024 um 08:46 Uhr

Persönliche Daten

Mag. Sandra Sportlich, MSc
Geschlecht:weiblich | geboren am: 24.8.1971 | SVNR:8567240856

Blutdruck

Zeitpunkt	Blutdruck	Notiz/Messbedingung	Messgerät (SNr.)
27. Januar 2024 um 07:48 Uhr	Systolisch: 115 (mmHg) Diastolisch: 67 (mmHg)	Maschinell	mg_bd1 (51204 00454)
28. Januar 2024 um 07:55 Uhr	Systolisch: 121 (mmHg) Diastolisch: 70 (mmHg) Puls: 79 (Schläge/min)	Maschinell	mg_bd1 (51204 00454)

Gewicht

Zeitpunkt	Gewicht	Notiz/Messbedingung	Messgerät (SNr.)
22. Januar 2024 um 09:05 Uhr	80.7 (kg)	Maschinell	mg_wg1 (40355 42306255)
28. Januar 2024 um 08:45 Uhr	80.6 (kg)		mg_wg1 (40355 42306255)

Schritte

Zeitpunkt	Schritte	Notiz/Messbedingung	Messgerät (SNr.)
20. Januar 2024 um 00:00 Uhr	6320 (pro Tag) --		mg_ft2 (3F7019550)
28. Januar 2024 um 00:00 Uhr	9800	Maschinell	mg_ft1 (73539 012302)

Blutzucker

Zeitpunkt	Blutzucker	Notiz/Messbedingung	Messgerät (SNr.)
27. Januar 2024 um 07:49 Uhr	201.5 (mg/dl) ++	nach dem Essen	mg_bz1 (12231 23128829)
28. Januar 2024 um 08:44 Uhr	155.5 (mg/dl)	nach dem Essen	mg_bz1 (12231 23128829)

Messgeräte

Gerätetyp	Hersteller	Modellnummer	Seriennummer
Blutdruck-Messgerät	A/D Medical	UA-767NFC	51204 00454
Waage	Wahoo Fitness	WFBTScalev1	40355 42306255
Fitness Tracker	Polar	loop-90047656	73539 012302
Fitness Tracker	Garmin	vivofit	3F7019550
Blutzuckermesser	Aktivmed	GlucoCheckXL	12231 23128829

Medikationsliste

Medikament	Einnahme	Dosierung	Hinweis	Zusatzinformation
Diazepam Actavis 10 mg Tabletten	täglich	1-0-1-0		Start: 2022-12-21 Ende: 2024-01-24 Anwendung: zum Einnehmen
Zithromax 250 mg Filmtabl.	täglich	1-0-0-0	vor dem Frühstück	Anwendung: zum Einnehmen
Nasivin 0,05 % - Nasentropfen	täglich		3x2 Tropfen	Anwendung: nasal
Seracril	bei Bedarf	1-1-1-0		Anwendung: zum Einnehmen

3 Legal

Die Ausarbeitung der Aufgabe wurde durch:

- OpenAI - GPT-4.5 Turbo,
- OpenAI - GPT-4.5 Vision,
- Anthropic - Claude 3 Opus,
- Kagi - FastGPT

mit mehreren unterschiedlichen Prompts und Custom Instructions sowie die verwendung des Ballmer Peak unterstützt.