



---

**Софийски университет „Св. Климент Охридски“ гр.  
София**

Факултет по математика и информатика

# ДОКУМЕНТАЦИЯ

към проект по „XML програмиране“ на тема:

„Фитнес център“

Изготвил:

Мария Стефанова Маргаритова, ФН: 1MI0700004, ИС 3 курс

Преподавател: Йоаннис Патиас 2023/2024 год.

# 1. Версия „XML схема“

## *a. Описание на предметната област и на заданието*

Проекта разглежда една примерна структура и организация на един фитнес център като включваме уредите, които се съдържат в него. Разглеждаме също видовете зали, които може да има в даден фитнес център, както и допълнителни удобства, които могат да бъдат включени в залите. В XML файловете се описват:

Фитнес(зали),

Залите(id, име, бележка, дали са само за жени, дали са само за мъже, дали има тоалетна, дали има климатик, каква е бройката на фитнес уредите,

уреди(име, минимални кг., максимални кг., настоящи кг.),

тренировка(дължина, дали е групова, за колко човека е, колко калории се горят, ритъм, брой почивки, дължина на почивката)).

## *b. Дефиниране на схемата*

```
<!ELEMENT gym (hall*)>
```

```
<!ELEMENT hall (id, name, note?, male_allowed, female_allowed, has_wc, has_ac, number_of equipments, equipment*, training)>
```

```
<!ELEMENT id (#PCDATA)>
```

```
<!ELEMENT name (#PCDATA)>
```

```
<!ELEMENT note (#PCDATA)>
```

```
<!ELEMENT male_allowed (#PCDATA)>
```

```
<!ELEMENT female_allowed (#PCDATA)>
```

```
<!ELEMENT has_wc (#PCDATA)>
```

```
<!ELEMENT has_ac (#PCDATA)>
```

```
<!ELEMENT number_of equipments (#PCDATA)>
```

```
<!ELEMENT equipment (name, min_kg, max_kg, current_kg)>
```

```
<!ELEMENT min_kg (#PCDATA)>
```

```
<!ELEMENT max_kg (#PCDATA)>
```

```
<!ELEMENT current_kg (#PCDATA)>
```

```
<!ELEMENT training (length, is_group, number_of_people?, calories_burnt, pace, number_of_breaks, break_length)>
```

```
<!ELEMENT length (#PCDATA)>
```

```
<!ELEMENT is_group (#PCDATA)>
```

<!ELEMENT number\_of\_people (#PCDATA)>

<!ELEMENT calories\_burnt (#PCDATA)>

<!ELEMENT pace (#PCDATA)>

<!ELEMENT number\_of\_breaks (#PCDATA)>

<!ELEMENT break\_length (#PCDATA)>

<!ATTLIST hall open (да | не) "да">

<!ATTLIST hall group\_training\_allowed (да | не) "да">

<!ATTLIST equipment type (машина | тежести) "тежести">

<!ATTLIST equipment machine\_type (бутаща | дърпаща) "бутаща">

<!ATTLIST training type (долна\_част | горна\_част | кардио | цяло\_тяло | корем) #REQUIRED>

## ● Описание на DTD файла

### Списък на елементите

- gym – основния елемент в XML документите, който може да съдържа елементи
- hall – елемент, обединяващ цялата подробна информация за залите, може да съдържа 0 или повече елемента
- id – елемент, номера на залата
- name – елемент, съдържащ символен низ, името на съответната зала
- note – елемент, съдържащ символен низ, с кратко описание на залата
- male\_allowed – елемент, съдържащ символен низ, показва позволено ли е влизането на мъже в съответната зала
- female\_allowed – елемент, съдържащ символен низ, показва позволено ли е влизането на жени в съответната зала
- has\_ws – елемент, съдържа информация за това дали има тоалетна в залата
- has\_ac – елемент, съдържа информация за това дали има климатик в залата
- number\_of equipments – елемент, съдържащ цяло число, броя на уредите
- equipment – елемент, съдържащ информация за уредите
- name – елемент, съдържащ символен низ, името на съответния уред
- min\_kg – елемент, съдържащ число, минималните допустими килограми на машината
- max\_kg – елемент, съдържащ число, максималните допустими килограми на машината
- current\_kg – елемент, съдържащ число, килограмите, които в момента са нагласени на машината
- training – елемент, съдържащ информация за тренировките
- length – елемент, съдържащ цяло число, дължина на съответната тренировка
- is\_group – елемент, съдържащ информация за това дали е групова дадената тренировка
- number\_of\_people – елемент, съдържащ цяло число, колко човека участват (ако не е групова е само един, ако е групова са повече)
- calories\_burnt – елемент, съдържащ цяло число, изгорени калории
- pace – елемент, съдържащ цяло число, показва темпото
- number\_of\_breaks – елемент, съдържащ цяло число, броя на почивките
- break\_length – елемент, съдържащ цяло число, колко дълга е една почивка

### Списък на атрибутите

- open – атрибут на елемента hall, показва дали залата е отворена в момента за ползване, да или не

- group-training\_allowed - атрибут на елемента hall, показва дали е позволена групов тренировка в тази зала, да или не.
- type – атрибут на елемента equipment, представящ типа на уреда и дали са тежести или машина
- machine\_type – атрибут на елемента equipment, представя типа на машината и дали е бутаща или дърпаща
- type – задължителен атрибут на елемента training, представя вида тренировка – долна част, горна част, кардио, цяло тяло, корем

### ***с. Добавяне на примерно съдържание, за доказване чрез примери на***

#### ***работоспособността на схемата***

В схемата по долу се вижда как изглежда един примерен валиден XML файл, който съответства на моята схема свързана с фитнес центъра.

```

▼<gym>
  ▼<hall open="да" group_training_allowed="да">
    <id>1</id>
    <name>Главна Зала</name>
    <note>Просторна зала с естествена светлина</note>
    <male_allowed>да</male_allowed>
    <female_allowed>да</female_allowed>
    <has_wc>да</has_wc>
    <has_ac>да</has_ac>
    <number_of equipments>10</number_of equipments>
    ▼<equipment type="тежести">
      <name>Дъмбели</name>
      <min_kg>5</min_kg>
      <max_kg>50</max_kg>
      <current_kg>10</current_kg>
    </equipment>
    ▼<training type="цяло_тяло">
      <length>60</length>
      <is_group>не</is_group>
      <calories_burnt>300</calories_burnt>
      <pace>средно</pace>
      <number_of_breaks>2</number_of_breaks>
      <break_length>5</break_length>
    </training>
  </hall>
</gym>

```

## 2. Версия „описание на DOM съответстващ на схемата“

### а. екземпляри на документа

Създадени са 5 екземпляра на документа. По-долу е пример за един от тях.

```
<gym>
  <hall open="да" group_training_allowed="да">
    <id>5</id>
    <name>Зала без Климатик</name>
    <male_allowed>да</male_allowed>
    <female_allowed>да</female_allowed>
    <has_wc>да</has_wc>
    <has_ac>не</has_ac>
    <number_of equipments>12</number_of equipments>
    <equipment type="тежести">
      <name>Кетълбел</name>
      <min_kg>5</min_kg>
      <max_kg>20</max_kg>
      <current_kg>12</current_kg>
    </equipment>
    <training type="цяло_тяло">
      <length>60</length>
      <is_group>не</is_group>
      <calories_burnt>350</calories_burnt>
      <pace>средно</pace>
      <number_of_breaks>2</number_of_breaks>
      <break_length>5</break_length>
    </training>
  </hall>
</gym>
```

### б. конструиране и валидиране на всеки екземпляр чрез DOM

За конструиране и валидиране на даден екземпляр съм използвала DOM като основните функции, които са употребени за тази цел са:

```
.createDocument(), .createHTMLDocument(),
doc.createElement(), .setAttribute(),
doc.createTextNode(), appendChild()
```

Благодарение на тях се създава едно дърво от всички елементи и атрибути и може да се използва файла много по-лесно.

Имаме 2 файла: main.js и xml.js., както и един, който се казва index.html. След като се отвори на екрана се появява бутон „Генерирай XML“. А при натисне – се появява генерирания XML.

### 3. Версия „набор от XSLT трансформации“

а. 2 различни стилови таблици (XSLT) за трансформиране на XML документите в друг XML код

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <html>
      <body>
        <h1 align="center">Фитнес</h1>
        <table border="3" align="center">
          <tr>
            <th>ID</th>
            <th>Име</th>
            <th>Информация</th>
          </tr>
          <xsl:for-each select="gym/hall">
            <tr>
              <td>
                <xsl:call-template name="id"/>
              </td>
              <td>
                <xsl:call-template name="name"/>
              </td>
              <td>
                <xsl:call-template name="note"/>
              </td>
            </tr>
          </xsl:for-each>
        </table>
      </body>
    </html>
  </xsl:template>

  <xsl:template name="id">
    <xsl:value-of select="id/."/>
  </xsl:template>

  <xsl:template name="name">
    <xsl:value-of select="name/."/>
  </xsl:template>

  <xsl:template name="note">
    <xsl:value-of select="note/."/>
  </xsl:template>
</xsl:stylesheet>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <html>
      <body>
        <h1 align="center">Тренировка</h1>
        <table border="3" align="center">
          <tr>
            <th>Дължина на тренировка</th>
            <th>Брой хора</th>
          </tr>
          <xsl:for-each select="gym/hall/training">
            <tr>
              <td>
                <xsl:call-template name="length"/>
              </td>
              <td>
                <xsl:call-template name="number_of_people"/>
              </td>
            </tr>
          </xsl:for-each>
        </table>
      </body>
    </html>
  </xsl:template>

  <xsl:template name="length">
    <xsl:value-of select="length/."/>
  </xsl:template>

  <xsl:template name="number_of_people">
    <xsl:value-of select="number_of_people/."/>
  </xsl:template>
</xsl:stylesheet>
```

ВИЗУАЛИЗАЦИЯ

## Тренировка

Дължина на тренировка	Брой хора
75	15

## Фитнес

ID	Име	Информация
1	Главна Зала	Просторна зала с естествена светлина

б. 2 различни стилови таблици (XSLT) за трансформиране на XML документите в чист текст

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="text"/>

  <xsl:template match="/">
    <xsl:for-each select="/gym/hall/training">

      Дължина: <xsl:value-of select="length"/>
      Брой почивки: <xsl:value-of select="number_of_breaks"/>

    </xsl:for-each>
  </xsl:template>
</xsl:stylesheet>
```

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="text"/>

  <xsl:template match="/">
    <xsl:for-each select="/gym/hall/training">

      ID: <xsl:value-of select="length"/>
      Име: <xsl:value-of select="name"/>
      Информация: <xsl:value-of select="note"/>

    </xsl:for-each>
  </xsl:template>
</xsl:stylesheet>
```

Дължина: 75  
Брой почивки: 3|

ID: 3  
Име: Зала за Групови Уроци  
Информация: Резервирана за групови занимания

## ВИЗУАЛИЗАЦИЯ

## 4. Заключение:

### *a. подобрения с добавяне на допълнителни атрибути*

Могат да се създадат най-различни носи атрибути, които да бъдат свързани с фитнес центъра и допълнителни удобства за посетителите, които да го правят по-привлекателен и посещаем: has\_minibar, has\_personal\_trainers, has\_spa, has\_pool, has\_tv, has\_discounts и тн.

### *b. доразвиване на схемата с добавяне на повече трансформации.*

Добавяйки повече трансформации, можем – при желание да създадем един уеб сайт на фитнес центъра. Като това е за един бъдещ проект, който може много да се доразвива. За него ще се използват html, css, js и трансформациите.