

## Solutions

Since we need to use different types of variables for a few different properties of an item, the struct structure was used. Name,surname, gender, email, phoneNumber ,address ,level\_of\_education,income\_level,expenditure ,currency\_unit ,current\_mood ,height ,weight variables were defined in the structure.

Using the typedef expression, the structure "struct \_element" was defined with the name "element".

Files with .dat extension as input files and .xml extensions as output files were sent to Bin2XML function. The data from the file was read into the tag and person elements.

The location information to be written has been sent to the createXML function. This function takes the name of the tag from the tag parameter, takes the data of that person from the person parameter and writes it as embedded.

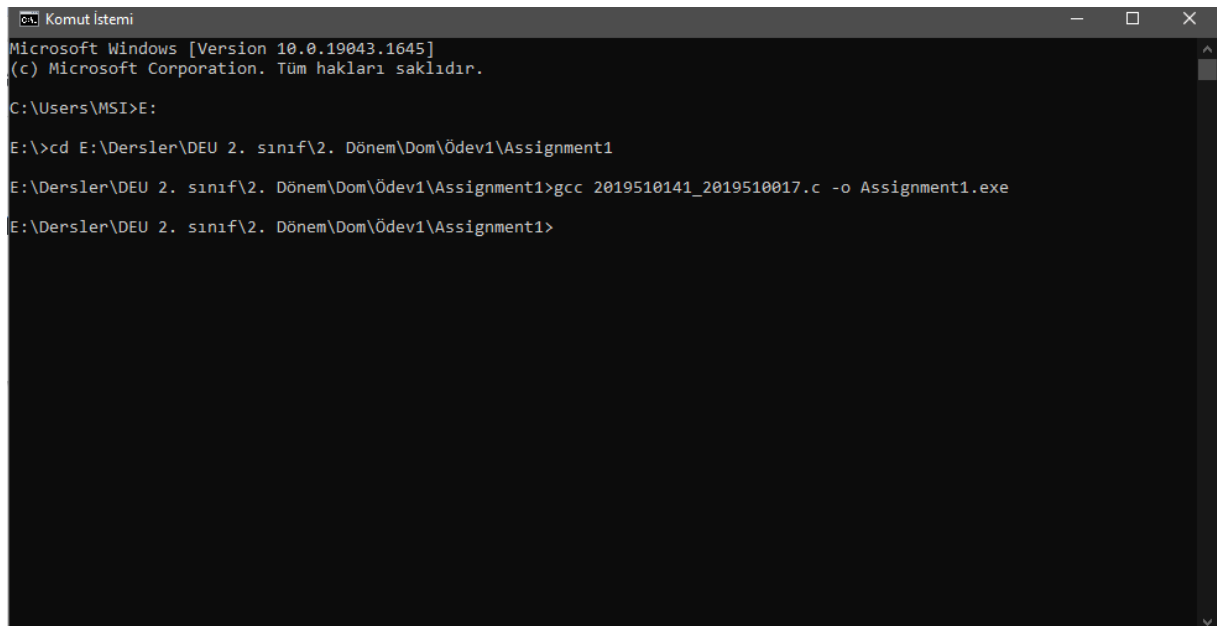
## Problems encountered

One of the major issues we ran into was when converting from little endian to big endian. we solved this problem with source code from <https://stackoverflow.com/questions/19275955/convert-little-endian-to-big-endian>.

We had to delete the emojis from the name section. For this, we first looked at the ASCII equivalents of all characters after conversion. Emoji characters had a value of -16. We checked the value of -16 in the code part and deleted the characters with this value. In this way, we have removed the emojis in the name part.

We had a hard time creating the XSD file. For this, we took help from the internet and used converters.

## Screenshots



```
Komut İstemi
Microsoft Windows [Version 10.0.19043.1645]
(c) Microsoft Corporation. Tüm hakları saklıdır.

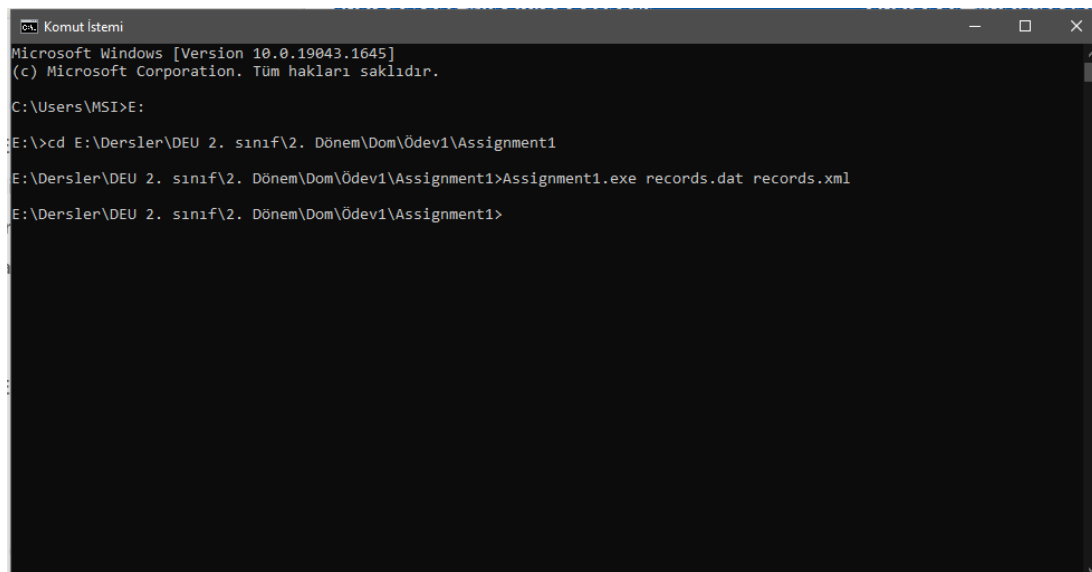
C:\Users\MSI>E:

E:\>cd E:\Dersler\DEU 2. sınıf\2. Dönem\Dom\Ödev1\Assignment1

E:\Dersler\DEU 2. sınıf\2. Dönem\Dom\Ödev1\Assignment1>gcc 2019510141_2019510017.c -o Assignment1.exe

E:\Dersler\DEU 2. sınıf\2. Dönem\Dom\Ödev1\Assignment1>
```

We used this command to convert the C code to exe.



```
Komut İstemi
Microsoft Windows [Version 10.0.19043.1645]
(c) Microsoft Corporation. Tüm hakları saklıdır.

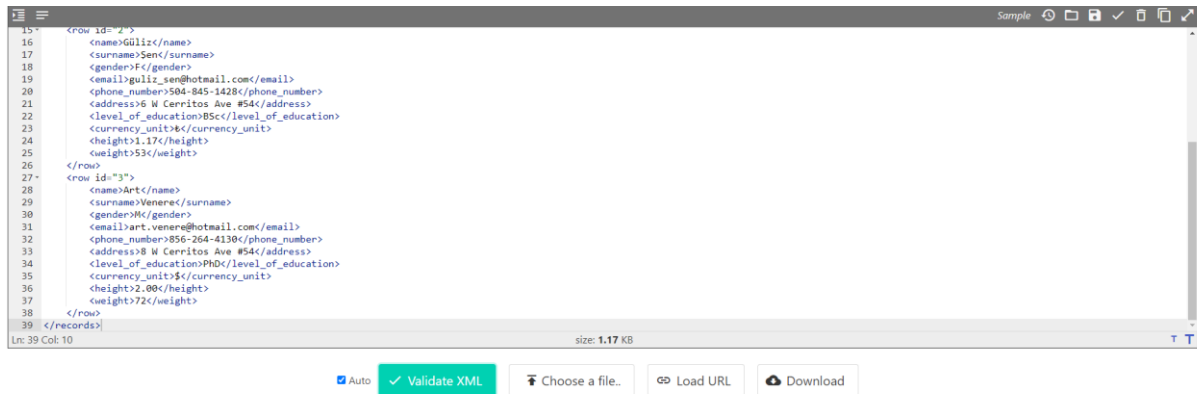
C:\Users\MSI>E:

E:\>cd E:\Dersler\DEU 2. sınıf\2. Dönem\Dom\Ödev1\Assignment1

E:\Dersler\DEU 2. sınıf\2. Dönem\Dom\Ödev1\Assignment1>Assignment1.exe records.dat records.xml

E:\Dersler\DEU 2. sınıf\2. Dönem\Dom\Ödev1\Assignment1>
```

This is how you should run the code via cmd.



Valid XML

We tried the Validate part from several sites available on the internet and it worked successfully.

Source is: <https://codebeautify.org/xmlvalidator>

```

2019510141_2019510017.c
1  #include <stdio.h>
2  #include <string.h>
3  #include <stdlib.h>
4
5  struct _element
6  {
7      char name[64];           // utf-16
8      char surname[32];       // utf-8
9      char gender;
10     char email[32];
11     char phoneNumber[16];
12     char address[32];
13     char levelOfEducation[8];
14     unsigned int incomeLevel; // given little-endian
15     unsigned int expenditure; // given big-endian
16     char currencyUnit[16];
17     char currentMood[32];
18     float height;
19     unsigned int weight;
20 };
21 typedef struct _element element;
22
23 unsigned int convertOrder(unsigned int toBeConverted)
24 {
25     int num = toBeConverted;
26     int b0,b1,b2,b3;
27     int res;
28
29     b0 = (num & 0x000000ff) << 24u;
30     b1 = (num & 0x0000ff00) << 8u;
31     b2 = (num & 0x00ff0000) >> 8u;
32     b3 = (num & 0xff000000) >> 24u;
33
34     res = b0 | b1 | b2 | b3;
35
36     return res;
37 }
38
39 void createXML(FILE *fb, element XMLelement, element tag)

```

Screenshot of part of the code.

```
<?xml version = "1.0" encoding = "utf-8"?>
<records>
  <row id="1">
    <name>James</name>
    <surname>Butt</surname>
    <gender>M</gender>
    <email>jbutt@gmail.com</email>
    <phone_number>504-845-1427</phone_number>
    <address>7 W Cerritos Ave #54</address>
    <level_of_education>MSc</level_of_education>
    <currency_unit>$</currency_unit>
    <height>1.33</height>
    <weight>68</weight>
  </row>
  <row id="2">
    <name>Güliz</name>
    <surname>Sen</surname>
    <gender>F</gender>
    <email>guliz_sen@hotmail.com</email>
    <phone_number>504-845-1428</phone_number>
    <address>6 W Cerritos Ave #54</address>
    <level_of_education>BSc</level_of_education>
    <currency_unit>₺</currency_unit>
    <height>1.17</height>
    <weight>53</weight>
  </row>
  <row id="3">
    <name>Art</name>
    <surname>Venere</surname>
    <gender>M</gender>
    <email>art.venere@hotmail.com</email>
    <phone_number>856-264-4130</phone_number>
    <address>8 W Cerritos Ave #54</address>
    <level_of_education>PhD</level_of_education>
    <currency_unit>$</currency_unit>
    <height>2.00</height>
    <weight>72</weight>
  </row>
</records>
```

Screenshot of the XML part.