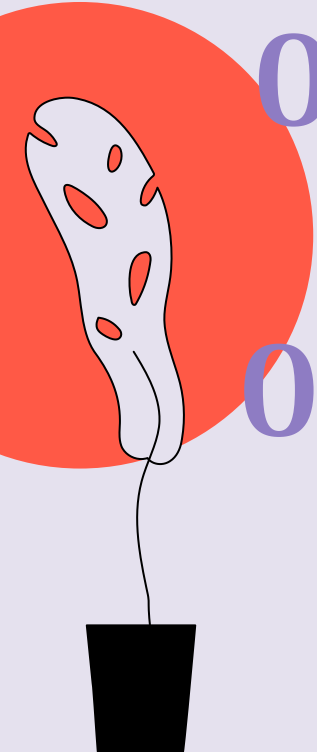
Storage and Data Recovery

Exercise 3: Convert data in XML or JSON format

Master's Degree in Intelligent Systems

Iván López Muñoz

Table of contents



01 Declare email

Represent 3 emails in XML format

02 Python program

Parse the XML file and show the results in the console

03 Format converter

Create a python program that converts XML format to JSON format

04 Final conversion

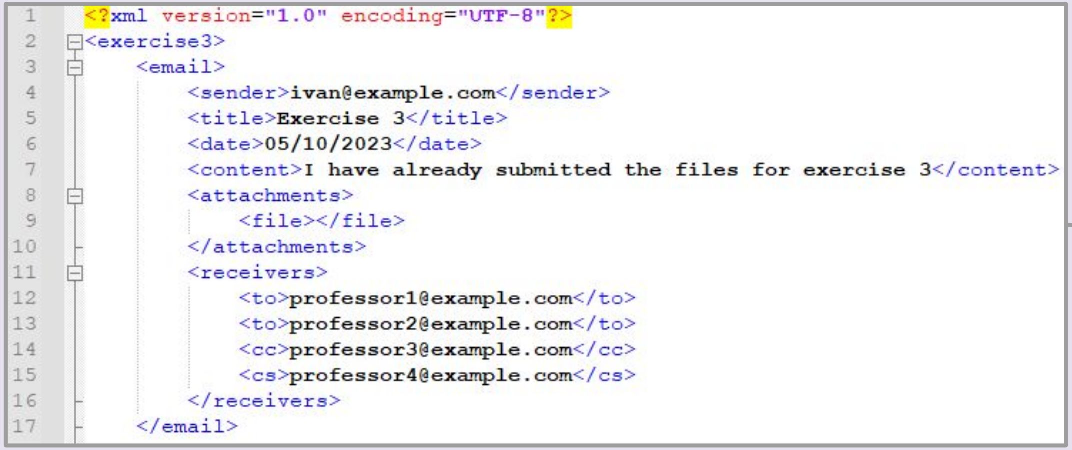
Finally, convert again the JSON format to XML and compare it with the former email file

01

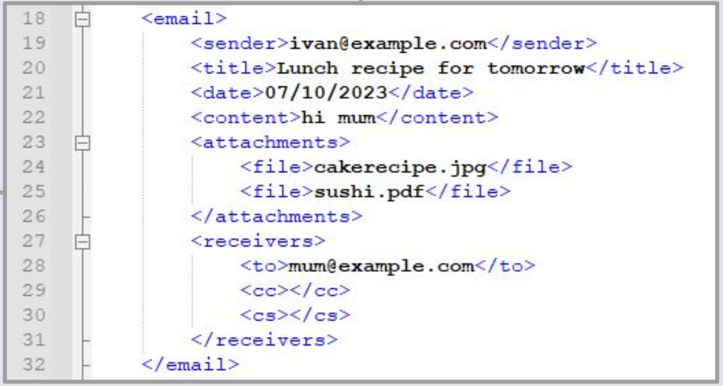
Declare former email

Represent 3 emails in XML format

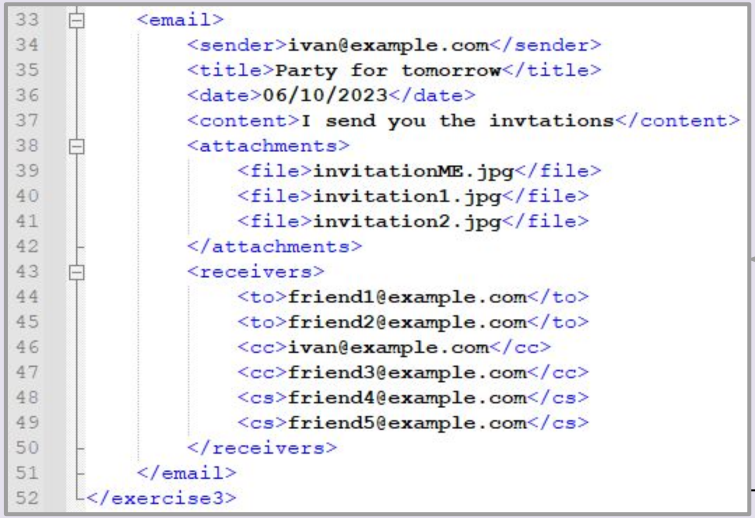
→



```xml
<?xml version="1.0" encoding="UTF-8"?>
<exercise3>
    <email>
        <sender>ivan@example.com</sender>
        <title>Exercise 3</title>
        <date>05/10/2023</date>
        <content>I have already submitted the files for exercise 3</content>
        <attachments>
            <file></file>
        </attachments>
        <receivers>
            <to>professor1@example.com</to>
            <to>professor2@example.com</to>
            <cc>professor3@example.com</cc>
            <cs>professor4@example.com</cs>
        </receivers>
    </email>
```



```xml
    <email>
        <sender>ivan@example.com</sender>
        <title>Lunch recipe for tomorrow</title>
        <date>07/10/2023</date>
        <content>hi mum</content>
        <attachments>
            <file>cakerecipe.jpg</file>
            <file>sushi.pdf</file>
        </attachments>
        <receivers>
            <to>mum@example.com</to>
            <cc></cc>
            <cs></cs>
        </receivers>
    </email>
```

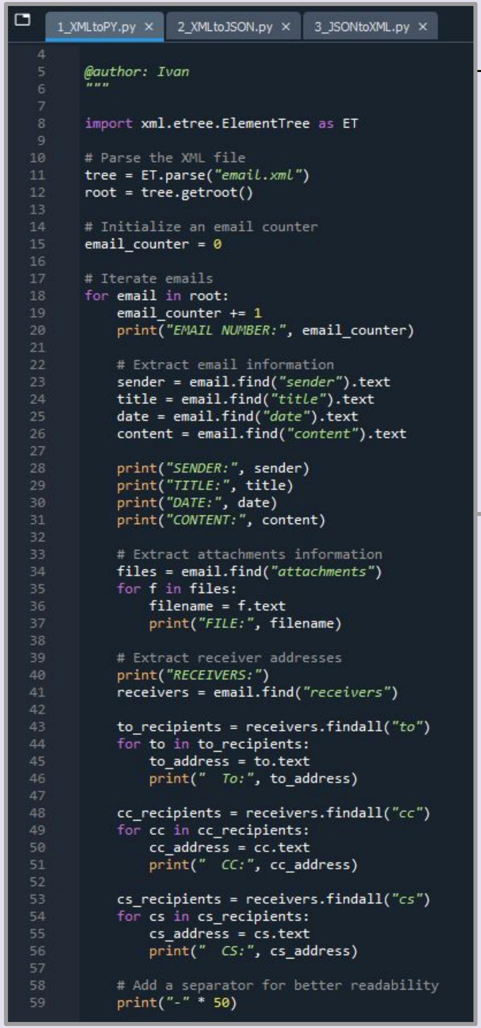


```xml
    <email>
        <sender>ivan@example.com</sender>
        <title>Party for tomorrow</title>
        <date>06/10/2023</date>
        <content>I send you the invtations</content>
        <attachments>
            <file>invitationME.jpg</file>
            <file>invitation1.jpg</file>
            <file>invitation2.jpg</file>
        </attachments>
        <receivers>
            <to>friend1@example.com</to>
            <to>friend2@example.com</to>
            <cc>ivan@example.com</cc>
            <cc>friend3@example.com</cc>
            <cs>friend4@example.com</cs>
            <cs>friend5@example.com</cs>
        </receivers>
    </email>
</exercise3>
```

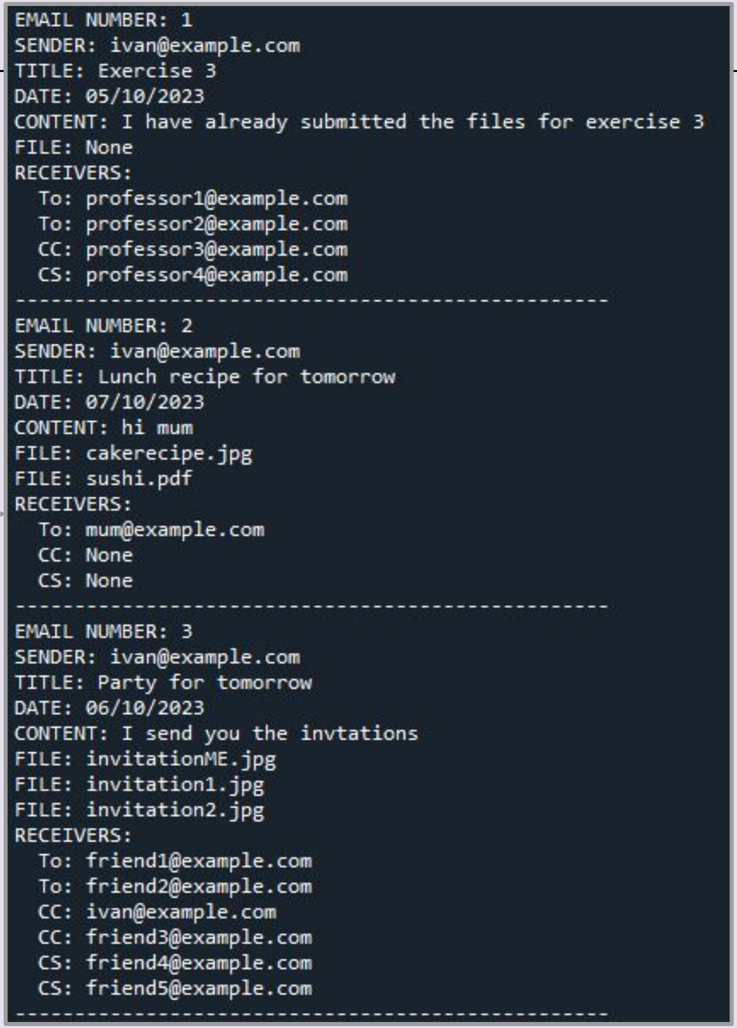
02

Python program

Parse the XML file and show the results in the console



```python

@author: Ivan
"""

import xml.etree.ElementTree as ET

# Parse the XML file
tree = ET.parse("email.xml")
root = tree.getroot()

# Initialize an email counter
email_counter = 0

# Iterate emails
for email in root:
    email_counter += 1
    print("EMAIL NUMBER:", email_counter)

    # Extract email information
    sender = email.find("sender").text
    title = email.find("title").text
    date = email.find("date").text
    content = email.find("content").text

    print("SENDER:", sender)
    print("TITLE:", title)
    print("DATE:", date)
    print("CONTENT:", content)

    # Extract attachments information
    files = email.find("attachments")
    for f in files:
        filename = f.text
        print("FILE:", filename)

    # Extract receiver addresses
    print("RECEIVERS:")
    receivers = email.find("receivers")

    to_recipients = receivers.findall("to")
    for to in to_recipients:
        to_address = to.text
        print("  To:", to_address)

    cc_recipients = receivers.findall("cc")
    for cc in cc_recipients:
        cc_address = cc.text
        print("  CC:", cc_address)

    cs_recipients = receivers.findall("cs")
    for cs in cs_recipients:
        cs_address = cs.text
        print("  CS:", cs_address)

    # Add a separator for better readability
    print("-" * 50)
```

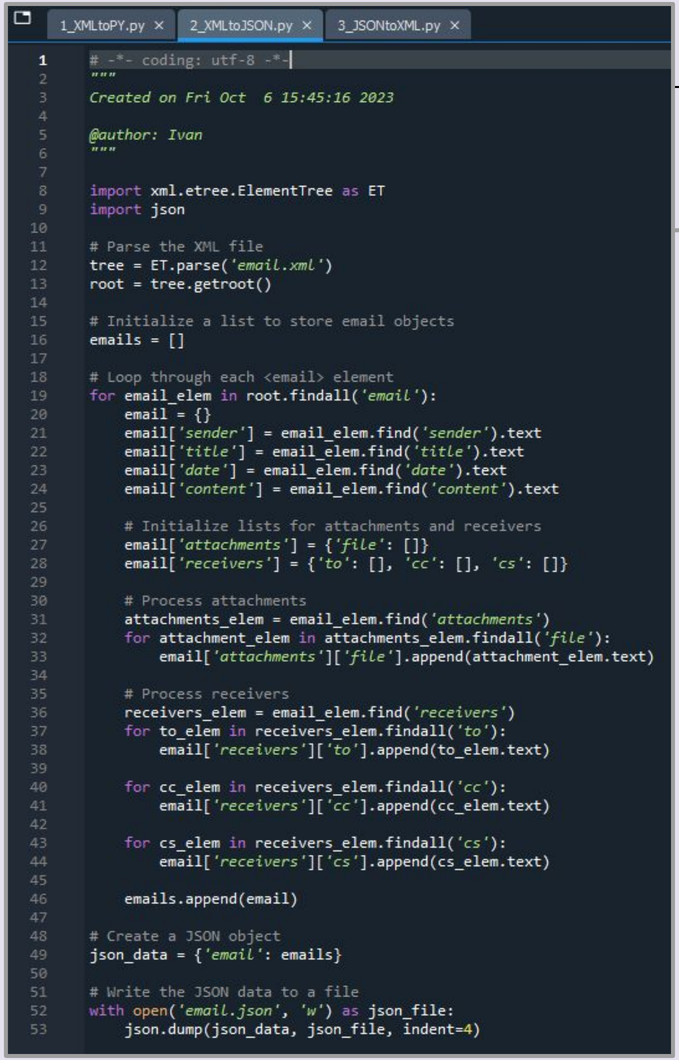


```
EMAIL NUMBER: 1
SENDER: ivan@example.com
TITLE: Exercise 3
DATE: 05/10/2023
CONTENT: I have already submitted the files for exercise 3
FILE: None
RECEIVERS:
  To: professor1@example.com
  To: professor2@example.com
  CC: professor3@example.com
  CS: professor4@example.com
--------------------------------------------------
EMAIL NUMBER: 2
SENDER: ivan@example.com
TITLE: Lunch recipe for tomorrow
DATE: 07/10/2023
CONTENT: hi mum
FILE: cakerecipe.jpg
FILE: sushi.pdf
RECEIVERS:
  To: mum@example.com
  CC: None
  CS: None
--------------------------------------------------
EMAIL NUMBER: 3
SENDER: ivan@example.com
TITLE: Party for tomorrow
DATE: 06/10/2023
CONTENT: I send you the invtations
FILE: invitationME.jpg
FILE: invitation1.jpg
FILE: invitation2.jpg
RECEIVERS:
  To: friend1@example.com
  To: friend2@example.com
  CC: ivan@example.com
  CC: friend3@example.com
  CS: friend4@example.com
  CS: friend5@example.com
--------------------------------------------------
```

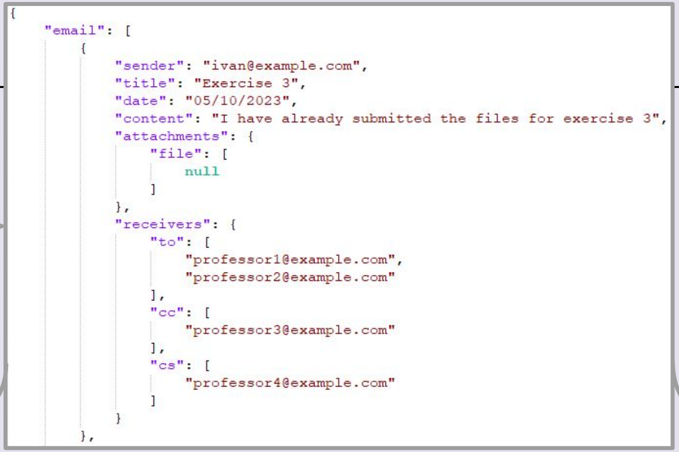
03

Format converter

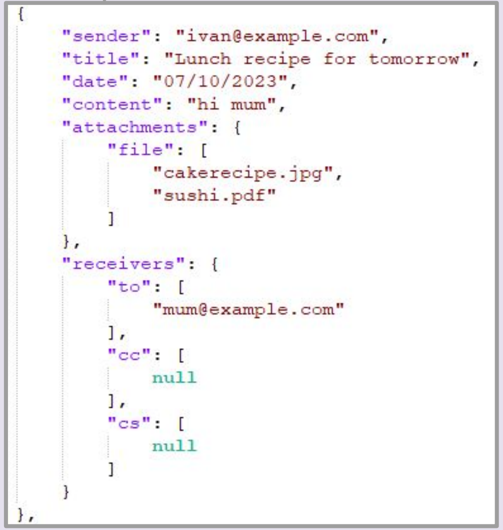
Create a python program that converts XML file to JSON format



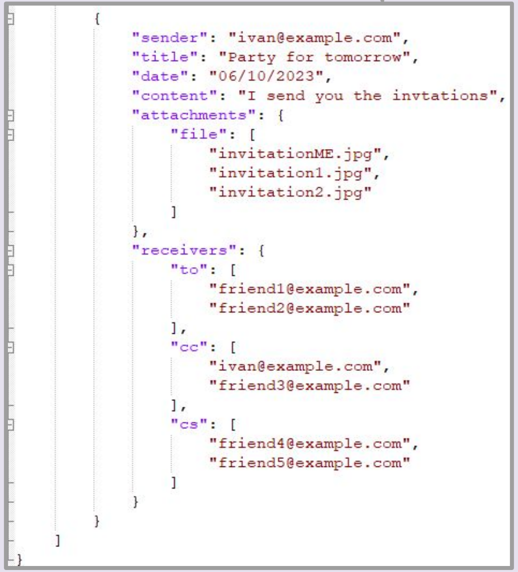
```python
# -*- coding: utf-8 -*-
"""
Created on Fri Oct  6 15:45:16 2023

@author: Ivan
"""

import xml.etree.ElementTree as ET
import json

# Parse the XML file
tree = ET.parse('email.xml')
root = tree.getroot()

# Initialize a list to store email objects
emails = []

# Loop through each <email> element
for email_elem in root.findall('email'):
    email = {}
    email['sender'] = email_elem.find('sender').text
    email['title'] = email_elem.find('title').text
    email['date'] = email_elem.find('date').text
    email['content'] = email_elem.find('content').text

    # Initialize lists for attachments and receivers
    email['attachments'] = {'file': []}
    email['receivers'] = {'to': [], 'cc': [], 'cs': []}

    # Process attachments
    attachments_elem = email_elem.find('attachments')
    for attachment_elem in attachments_elem.findall('file'):
        email['attachments']['file'].append(attachment_elem.text)

    # Process receivers
    receivers_elem = email_elem.find('receivers')
    for to_elem in receivers_elem.findall('to'):
        email['receivers']['to'].append(to_elem.text)

    for cc_elem in receivers_elem.findall('cc'):
        email['receivers']['cc'].append(cc_elem.text)

    for cs_elem in receivers_elem.findall('cs'):
        email['receivers']['cs'].append(cs_elem.text)

    emails.append(email)

# Create a JSON object
json_data = {'email': emails}

# Write the JSON data to a file
with open('email.json', 'w') as json_file:
    json.dump(json_data, json_file, indent=4)
```



```json
{
    "email": [
        {
            "sender": "ivan@example.com",
            "title": "Exercise 3",
            "date": "05/10/2023",
            "content": "I have already submitted the files for exercise 3",
            "attachments": {
                "file": [
                    null
                ]
            },
            "receivers": {
                "to": [
                    "professor1@example.com",
                    "professor2@example.com"
                ],
                "cc": [
                    "professor3@example.com"
                ],
                "cs": [
                    "professor4@example.com"
                ]
            }
        },
```



```json
{
    "sender": "ivan@example.com",
    "title": "Lunch recipe for tomorrow",
    "date": "07/10/2023",
    "content": "hi mum",
    "attachments": {
        "file": [
            "cakerecipe.jpg",
            "sushi.pdf"
        ]
    },
    "receivers": {
        "to": [
            "mum@example.com"
        ],
        "cc": [
            null
        ],
        "cs": [
            null
        ]
    }
},
```



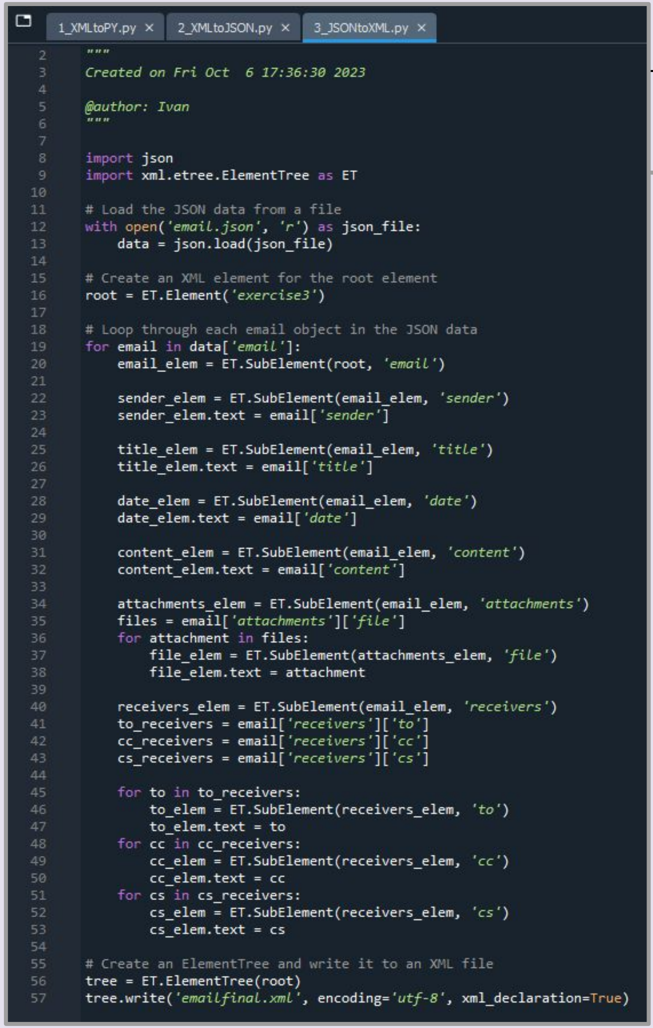
```json
        {
            "sender": "ivan@example.com",
            "title": "Party for tomorrow",
            "date": "06/10/2023",
            "content": "I send you the invtations",
            "attachments": {
                "file": [
                    "invitationME.jpg",
                    "invitation1.jpg",
                    "invitation2.jpg"
                ]
            },
            "receivers": {
                "to": [
                    "friend1@example.com",
                    "friend2@example.com"
                ],
                "cc": [
                    "ivan@example.com",
                    "friend3@example.com"
                ],
                "cs": [
                    "friend4@example.com",
                    "friend5@example.com"
                ]
            }
        }
    ]
}
```

04

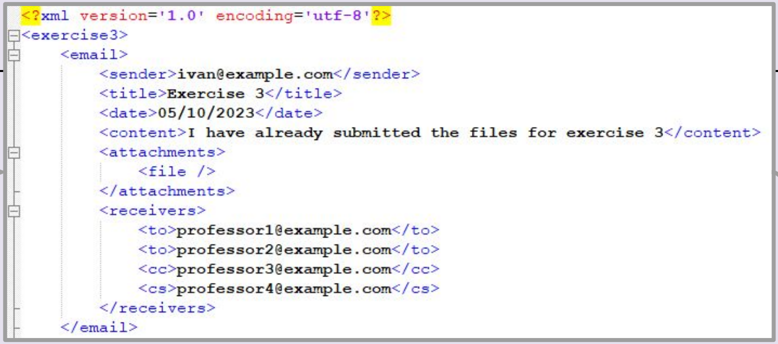
Final conversion

Finally, convert again the JSON file to XML and compare it with the former email file

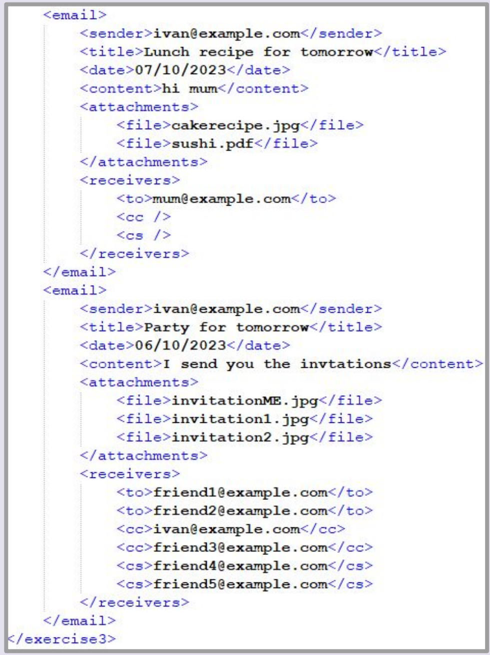
→



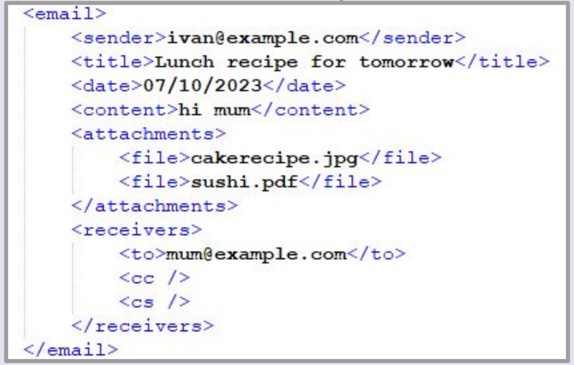
```python
"""
Created on Fri Oct  6 17:36:30 2023

@author: Ivan
"""

import json
import xml.etree.ElementTree as ET

# Load the JSON data from a file
with open('email.json', 'r') as json_file:
    data = json.load(json_file)

# Create an XML element for the root element
root = ET.Element('exercise3')

# Loop through each email object in the JSON data
for email in data['email']:
    email_elem = ET.SubElement(root, 'email')

    sender_elem = ET.SubElement(email_elem, 'sender')
    sender_elem.text = email['sender']

    title_elem = ET.SubElement(email_elem, 'title')
    title_elem.text = email['title']

    date_elem = ET.SubElement(email_elem, 'date')
    date_elem.text = email['date']

    content_elem = ET.SubElement(email_elem, 'content')
    content_elem.text = email['content']

    attachments_elem = ET.SubElement(email_elem, 'attachments')
    files = email['attachments']['file']
    for attachment in files:
        file_elem = ET.SubElement(attachments_elem, 'file')
        file_elem.text = attachment

    receivers_elem = ET.SubElement(email_elem, 'receivers')
    to_receivers = email['receivers']['to']
    cc_receivers = email['receivers']['cc']
    cs_receivers = email['receivers']['cs']

    for to in to_receivers:
        to_elem = ET.SubElement(receivers_elem, 'to')
        to_elem.text = to
    for cc in cc_receivers:
        cc_elem = ET.SubElement(receivers_elem, 'cc')
        cc_elem.text = cc
    for cs in cs_receivers:
        cs_elem = ET.SubElement(receivers_elem, 'cs')
        cs_elem.text = cs

# Create an ElementTree and write it to an XML file
tree = ET.ElementTree(root)
tree.write('emailfinal.xml', encoding='utf-8', xml_declaration=True)
```



```xml
<?xml version='1.0' encoding='utf-8'?>
<exercise3>
    <email>
        <sender>ivan@example.com</sender>
        <title>Exercise 3</title>
        <date>05/10/2023</date>
        <content>I have already submitted the files for exercise 3</content>
        <attachments>
            <file />
        </attachments>
        <receivers>
            <to>professor1@example.com</to>
            <to>professor2@example.com</to>
            <cc>professor3@example.com</cc>
            <cs>professor4@example.com</cs>
        </receivers>
    </email>
```



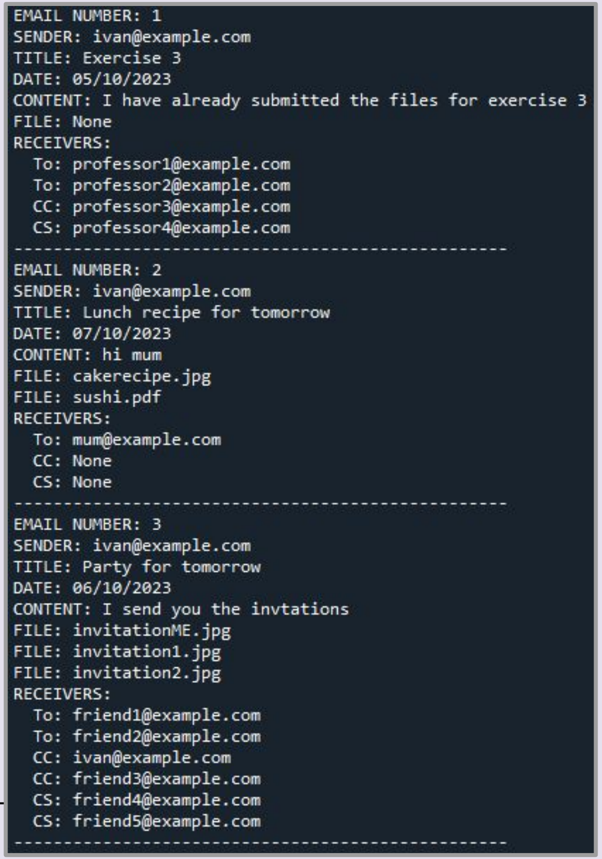
```xml
    <email>
        <sender>ivan@example.com</sender>
        <title>Lunch recipe for tomorrow</title>
        <date>07/10/2023</date>
        <content>hi mum</content>
        <attachments>
            <file>cakerecipe.jpg</file>
            <file>sushi.pdf</file>
        </attachments>
        <receivers>
            <to>mum@example.com</to>
            <cc />
            <cs />
        </receivers>
    </email>
    <email>
        <sender>ivan@example.com</sender>
        <title>Party for tomorrow</title>
        <date>06/10/2023</date>
        <content>I send you the invtations</content>
        <attachments>
            <file>invitationME.jpg</file>
            <file>invitation1.jpg</file>
            <file>invitation2.jpg</file>
        </attachments>
        <receivers>
            <to>friend1@example.com</to>
            <to>friend2@example.com</to>
            <cc>ivan@example.com</cc>
            <cc>friend3@example.com</cc>
            <cs>friend4@example.com</cs>
            <cs>friend5@example.com</cs>
        </receivers>
    </email>
</exercise3>
```



```xml
<email>
    <sender>ivan@example.com</sender>
    <title>Lunch recipe for tomorrow</title>
    <date>07/10/2023</date>
    <content>hi mum</content>
    <attachments>
        <file>cakerecipe.jpg</file>
        <file>sushi.pdf</file>
    </attachments>
    <receivers>
        <to>mum@example.com</to>
        <cc />
        <cs />
    </receivers>
</email>
```

File comparison

To compare the results of the former and final file, we are going to use the final XML file generated in the first python program to see through the console if there is any difference:



```
EMAIL NUMBER: 1
SENDER: ivan@example.com
TITLE: Exercise 3
DATE: 05/10/2023
CONTENT: I have already submitted the files for exercise 3
FILE: None
RECEIVERS:
  To: professor1@example.com
  To: professor2@example.com
  CC: professor3@example.com
  CS: professor4@example.com
--------------------------------------------------
EMAIL NUMBER: 2
SENDER: ivan@example.com
TITLE: Lunch recipe for tomorrow
DATE: 07/10/2023
CONTENT: hi mum
FILE: cakerecipe.jpg
FILE: sushi.pdf
RECEIVERS:
  To: mum@example.com
  CC: None
  CS: None
--------------------------------------------------
EMAIL NUMBER: 3
SENDER: ivan@example.com
TITLE: Party for tomorrow
DATE: 06/10/2023
CONTENT: I send you the invtations
FILE: invitationME.jpg
FILE: invitation1.jpg
FILE: invitation2.jpg
RECEIVERS:
  To: friend1@example.com
  To: friend2@example.com
  CC: ivan@example.com
  CC: friend3@example.com
  CS: friend4@example.com
  CS: friend5@example.com
--------------------------------------------------
```

As we can see the result is the same. The only difference is the way the format is written, when an element is null, instead of writing <file></file>, python writes it as <file />