

AHMED'S PORTFOLIO



Click or Scan QR Code

Geographic
Information
System
Analyst

✉ ahmedmohamed24997@gmail.com

📞 +966539826756

📞 +966580205720

📍 Riyadh, Saudi Arabia



Portfolio
Website



Download
CV



Download
Certificates



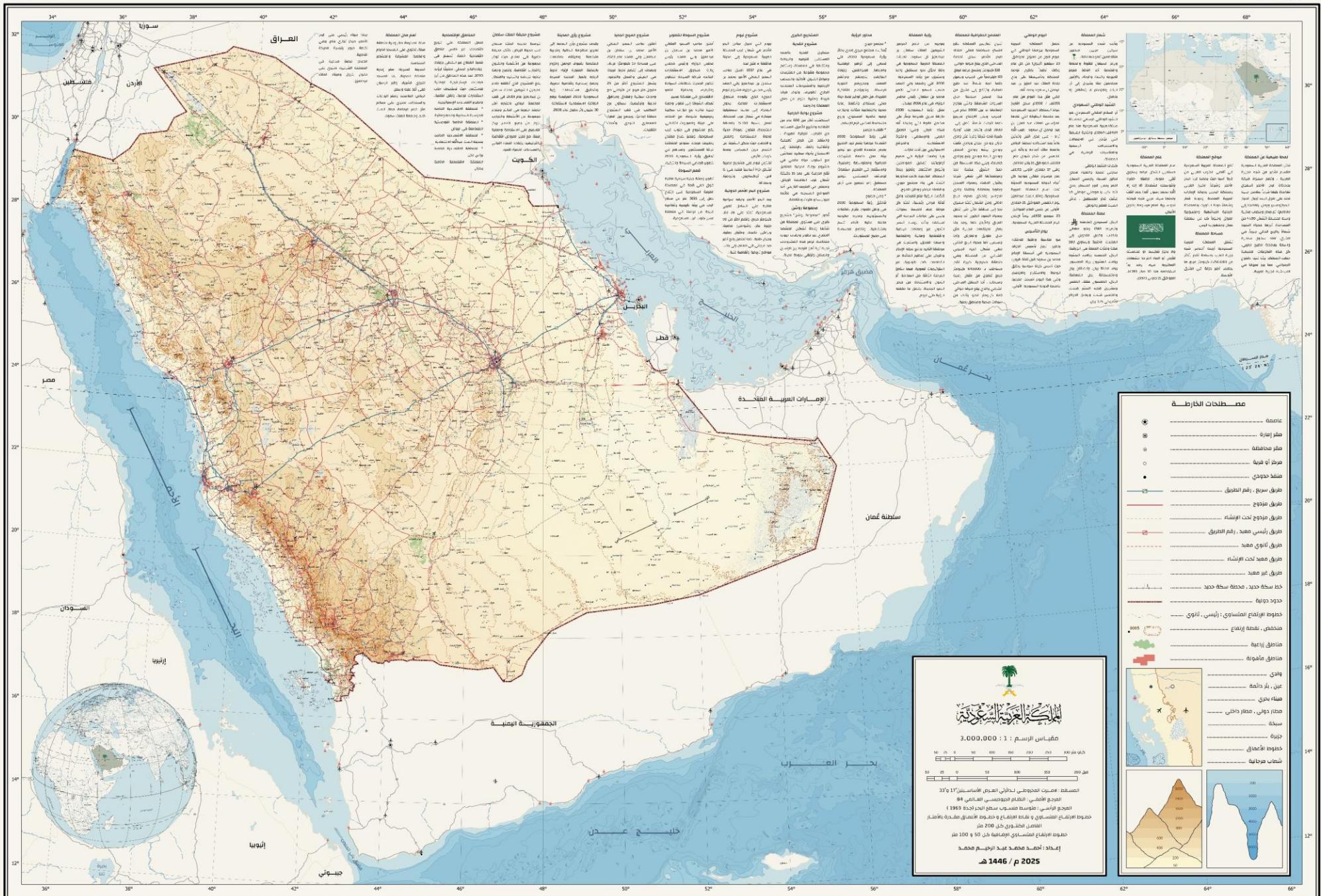
Download
Maps



About Me

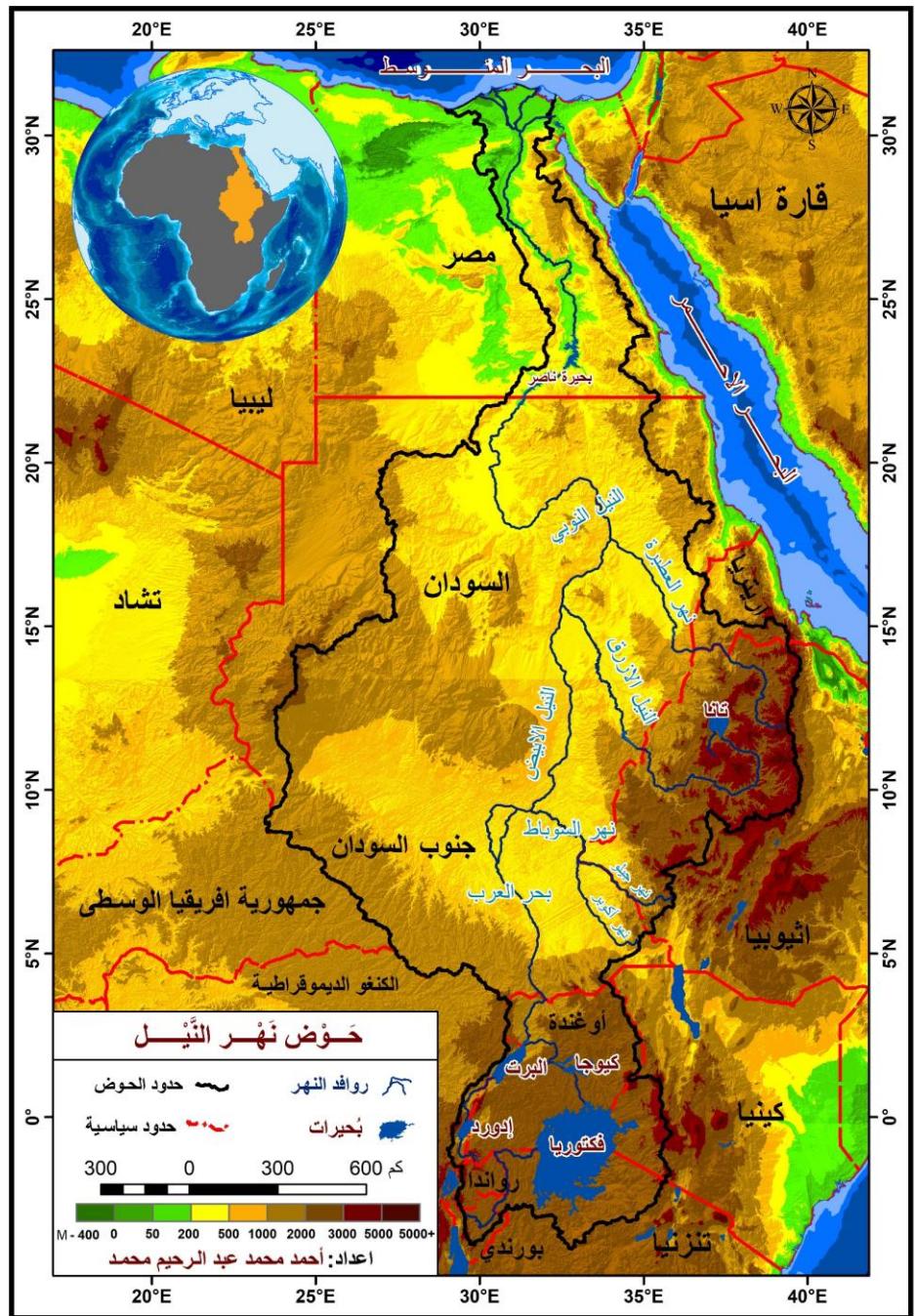


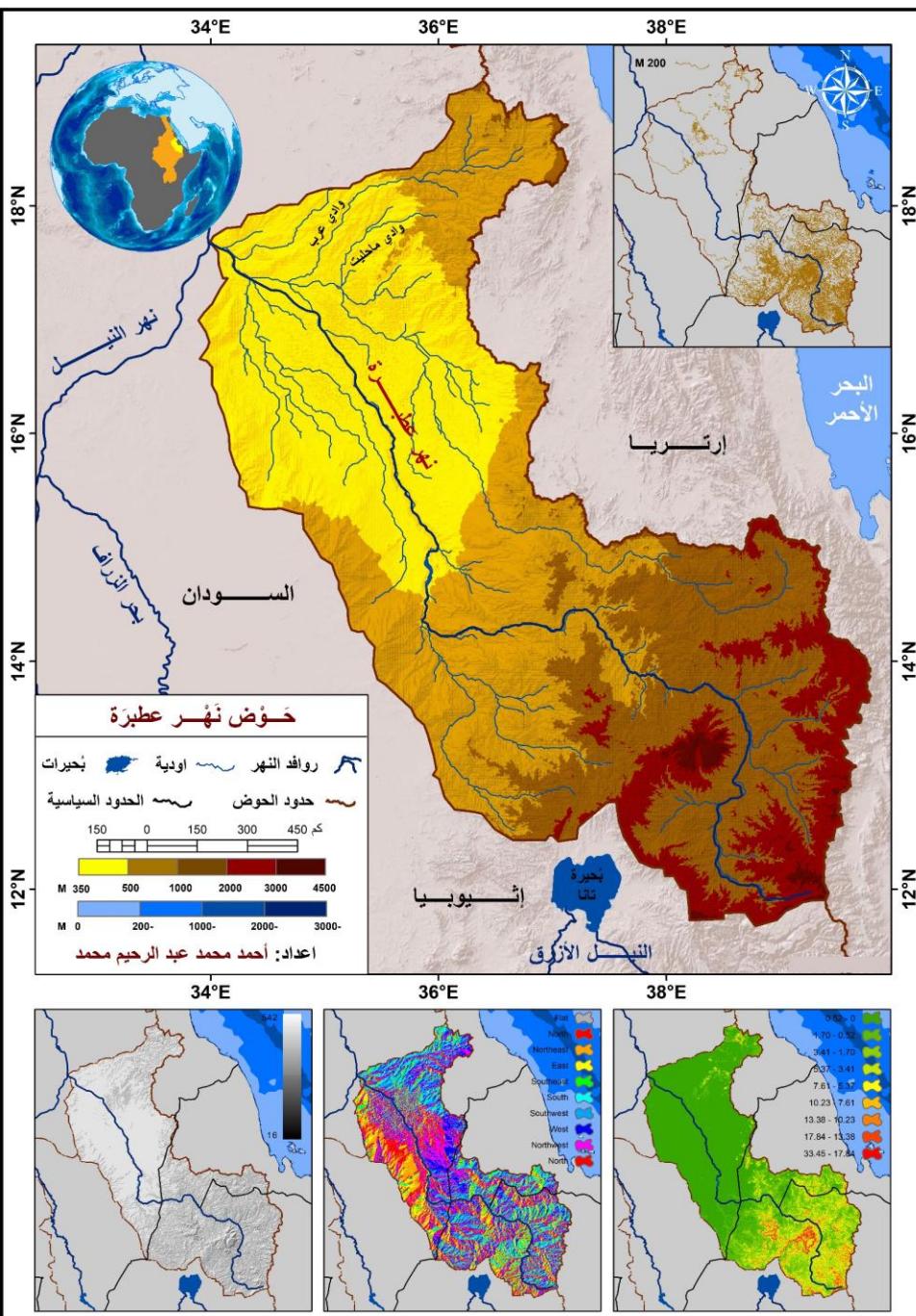
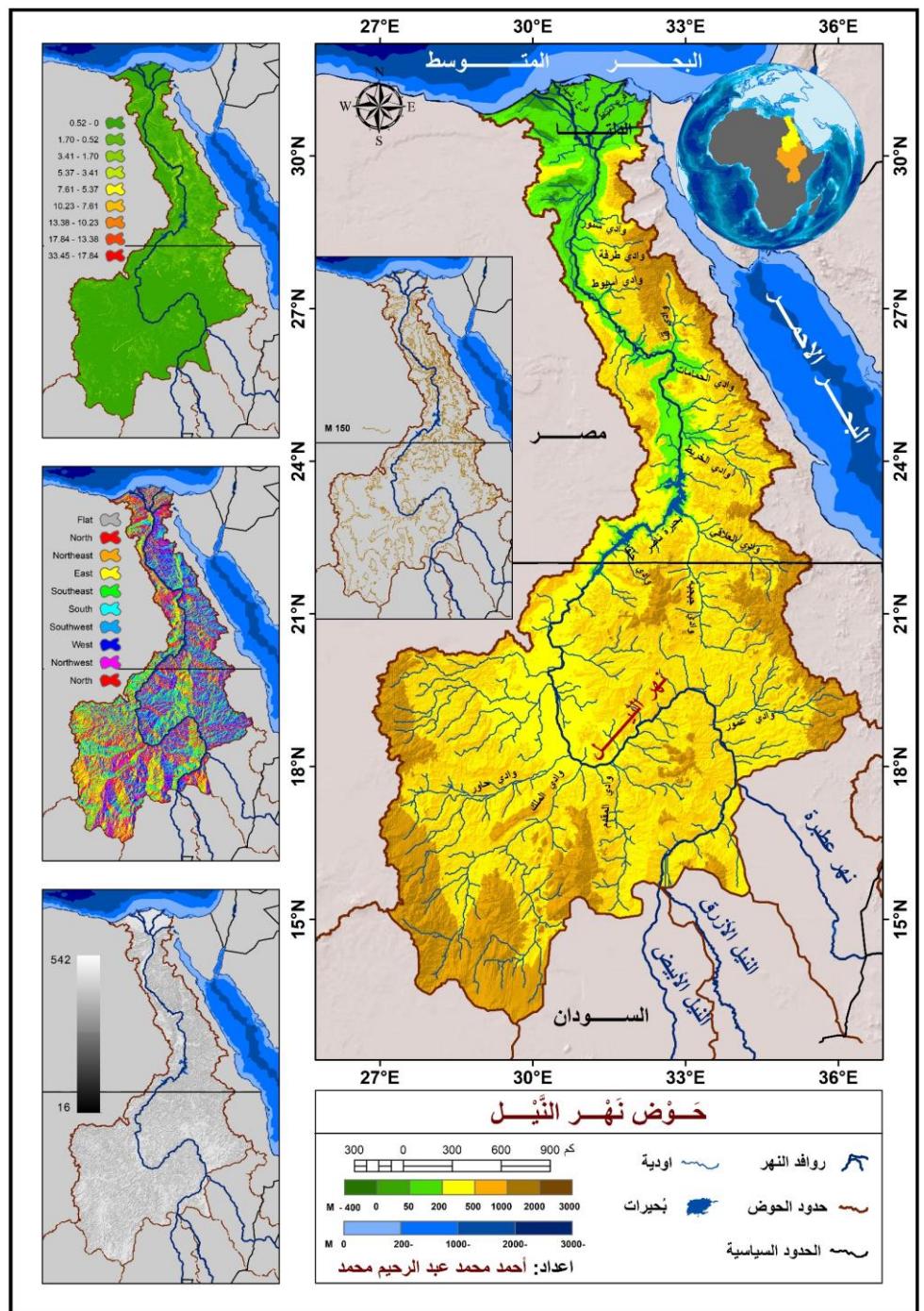
Results-driven GIS Specialist and Web Developer with over 6 years of experience delivering innovative geospatial solutions and dynamic web applications for government and private sector clients. Highly skilled in collecting, analyzing, and managing spatial data to produce high-quality maps, dashboards, and spatial analyses. Proficient in ArcGIS Pro, ArcGIS Online, and Python scripting for automating GIS workflows, with strong frontend development capabilities using HTML, CSS, JavaScript, Bootstrap, and Tailwind CSS to create responsive, intuitive, and user-friendly websites. Experienced in asset inventory projects, field data collection, and multiuser geodatabase management. Passionate about integrating emerging technologies to optimize geospatial operations and develop impactful web-based solutions.

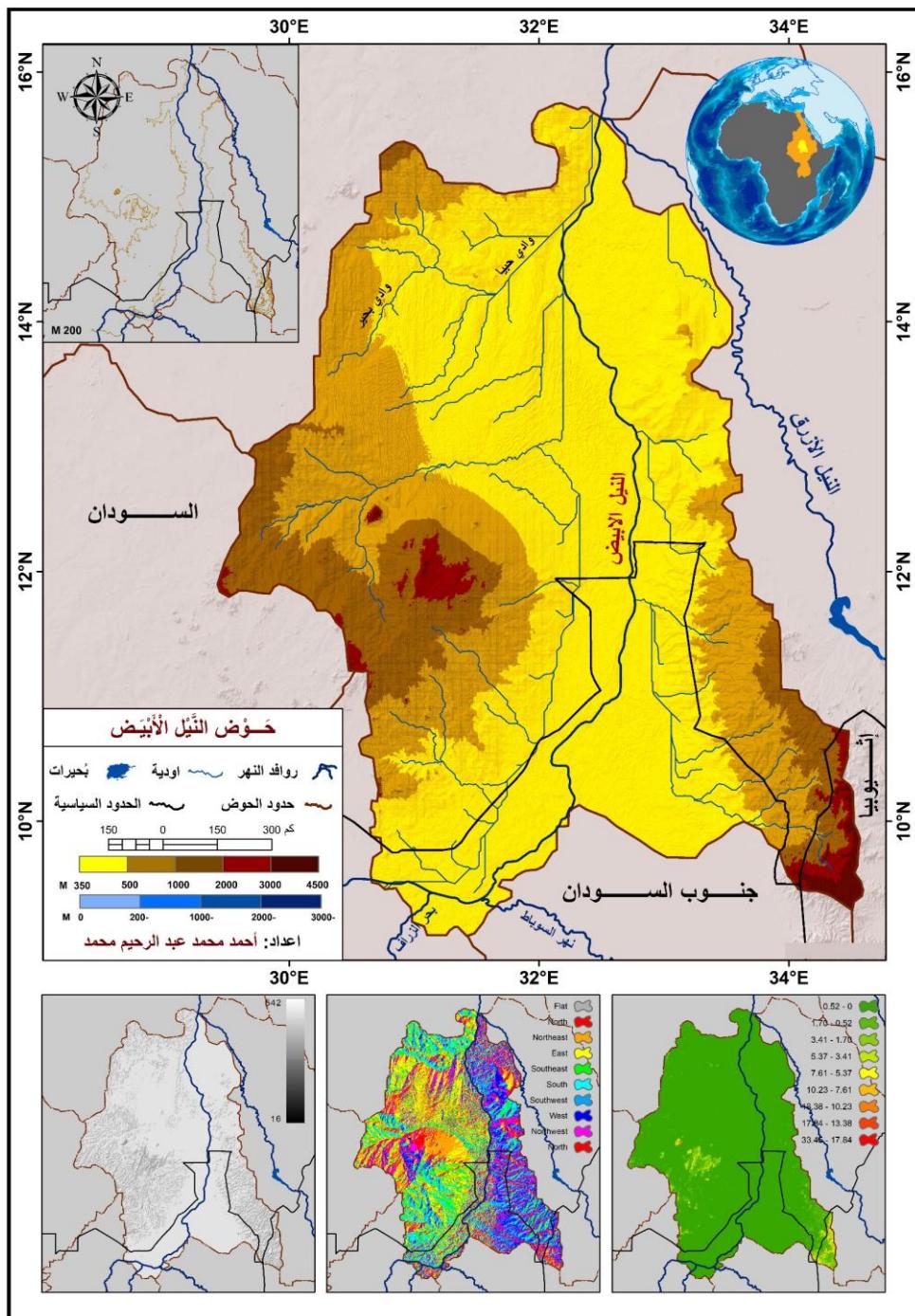
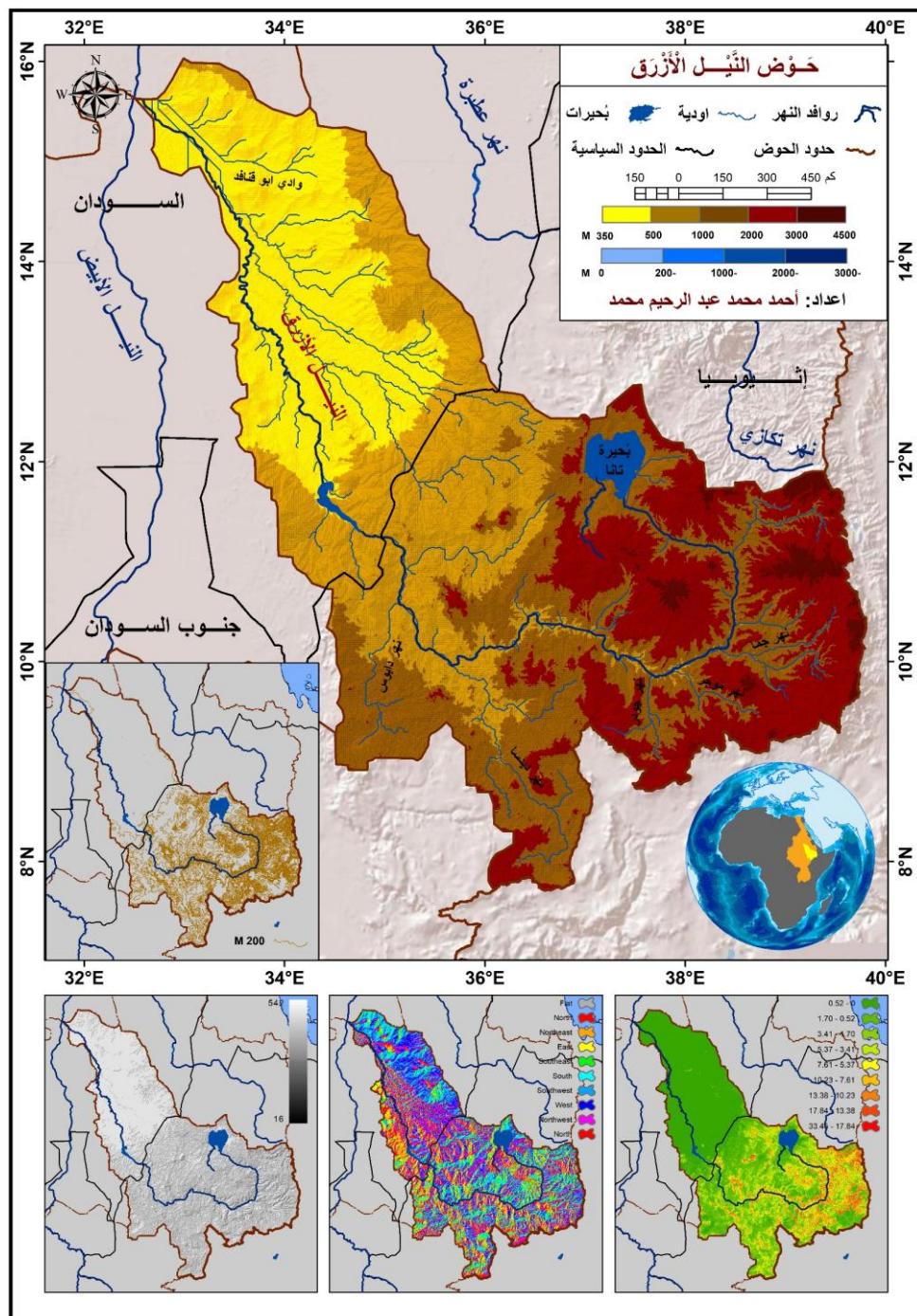


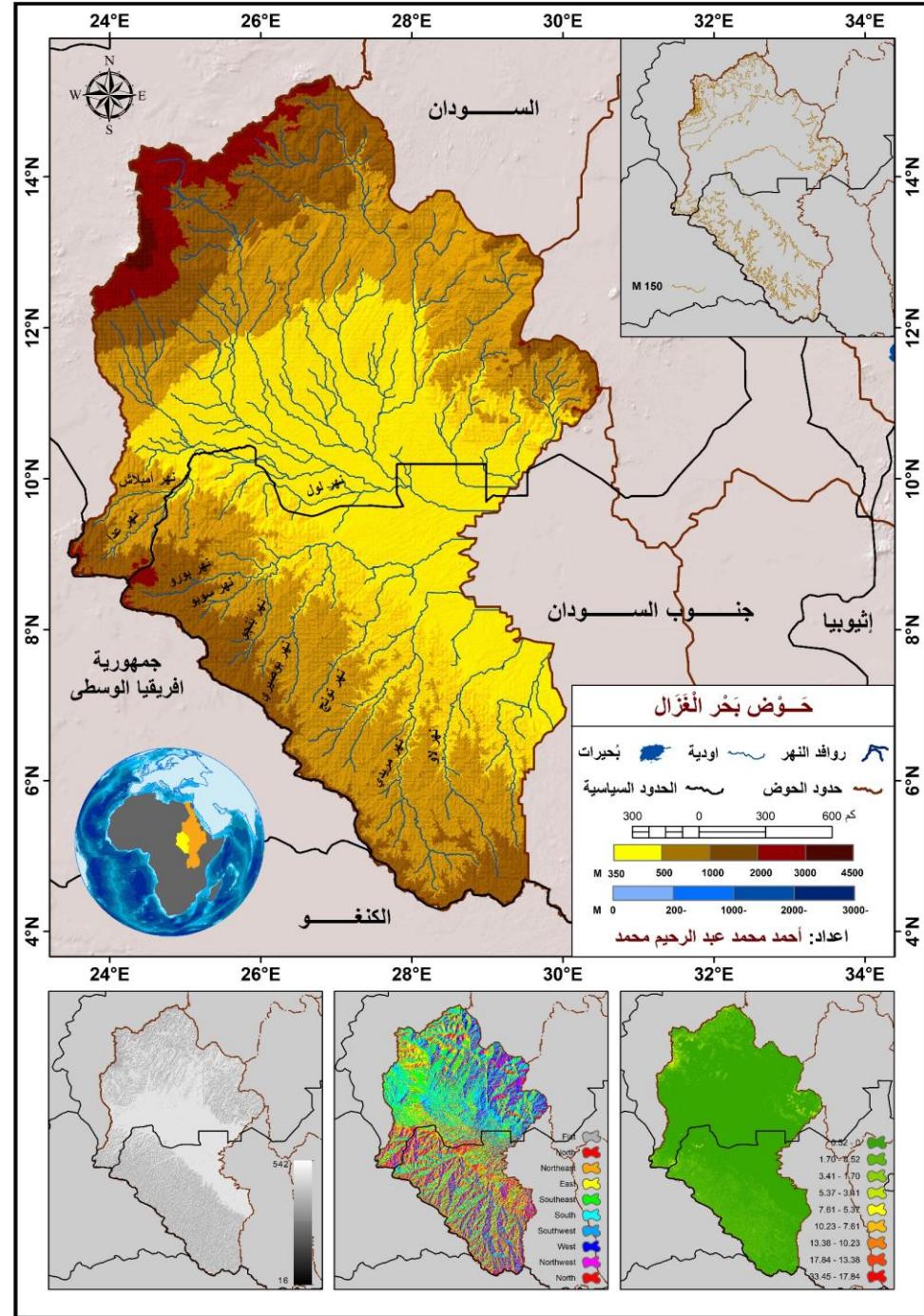
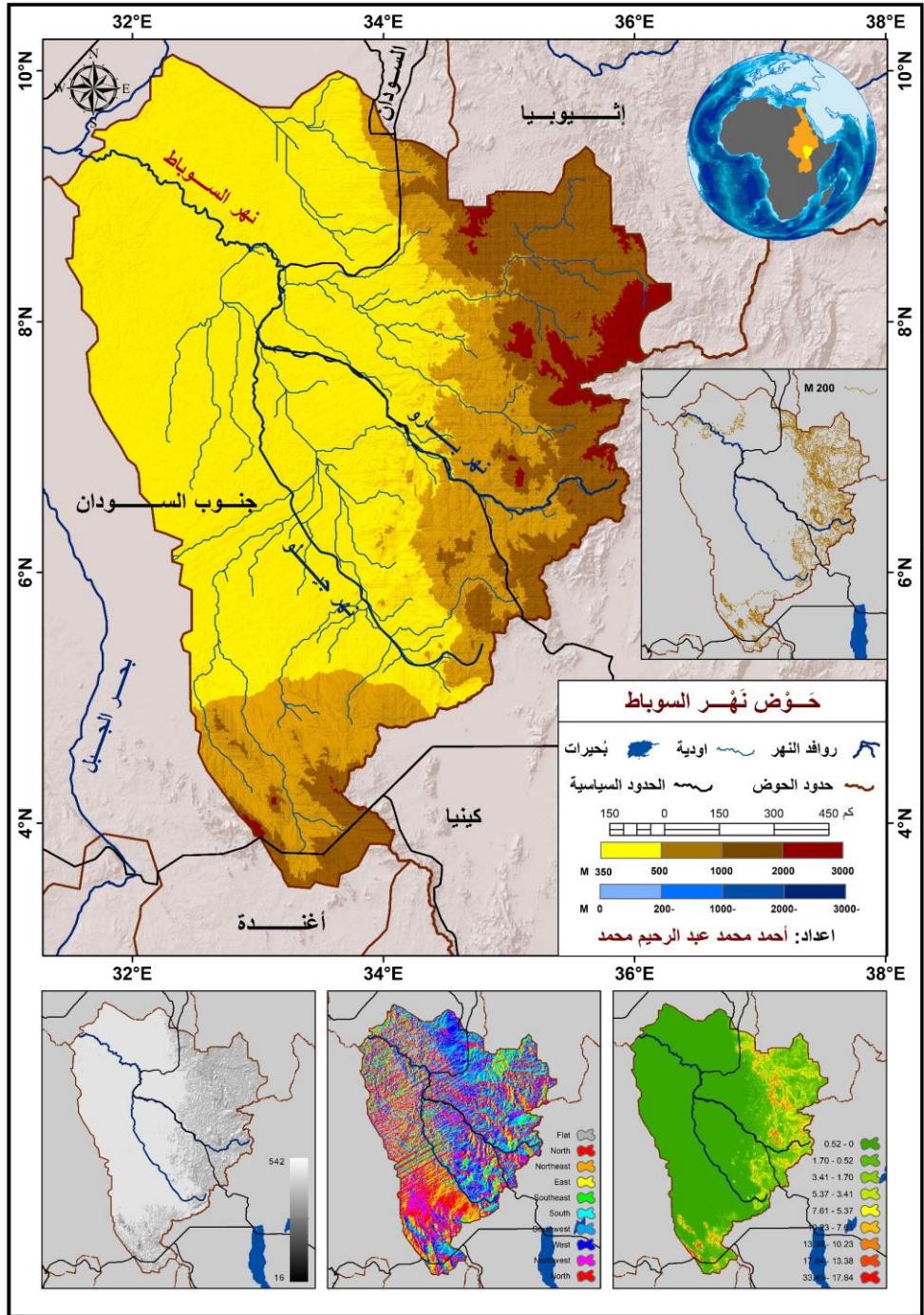


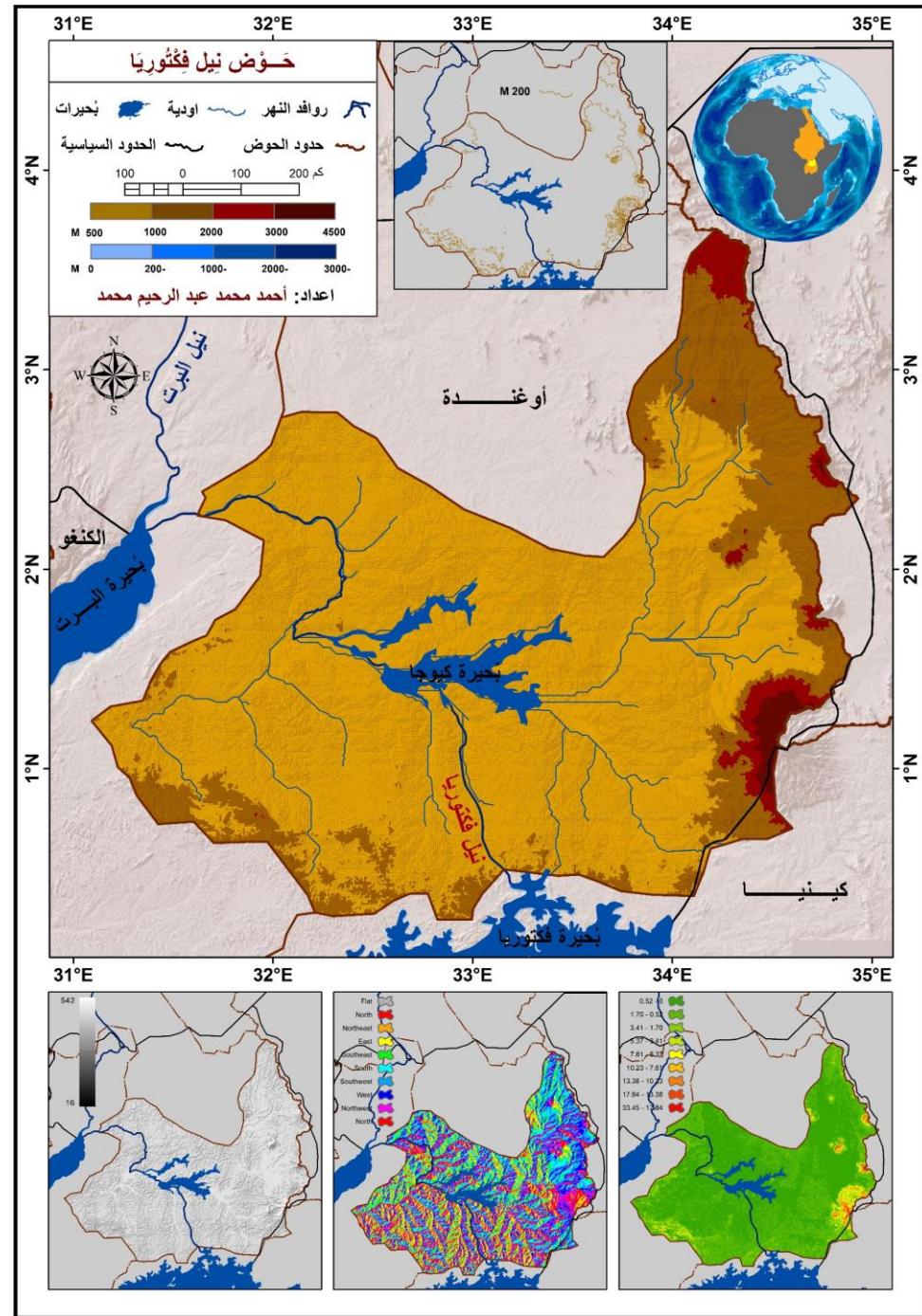
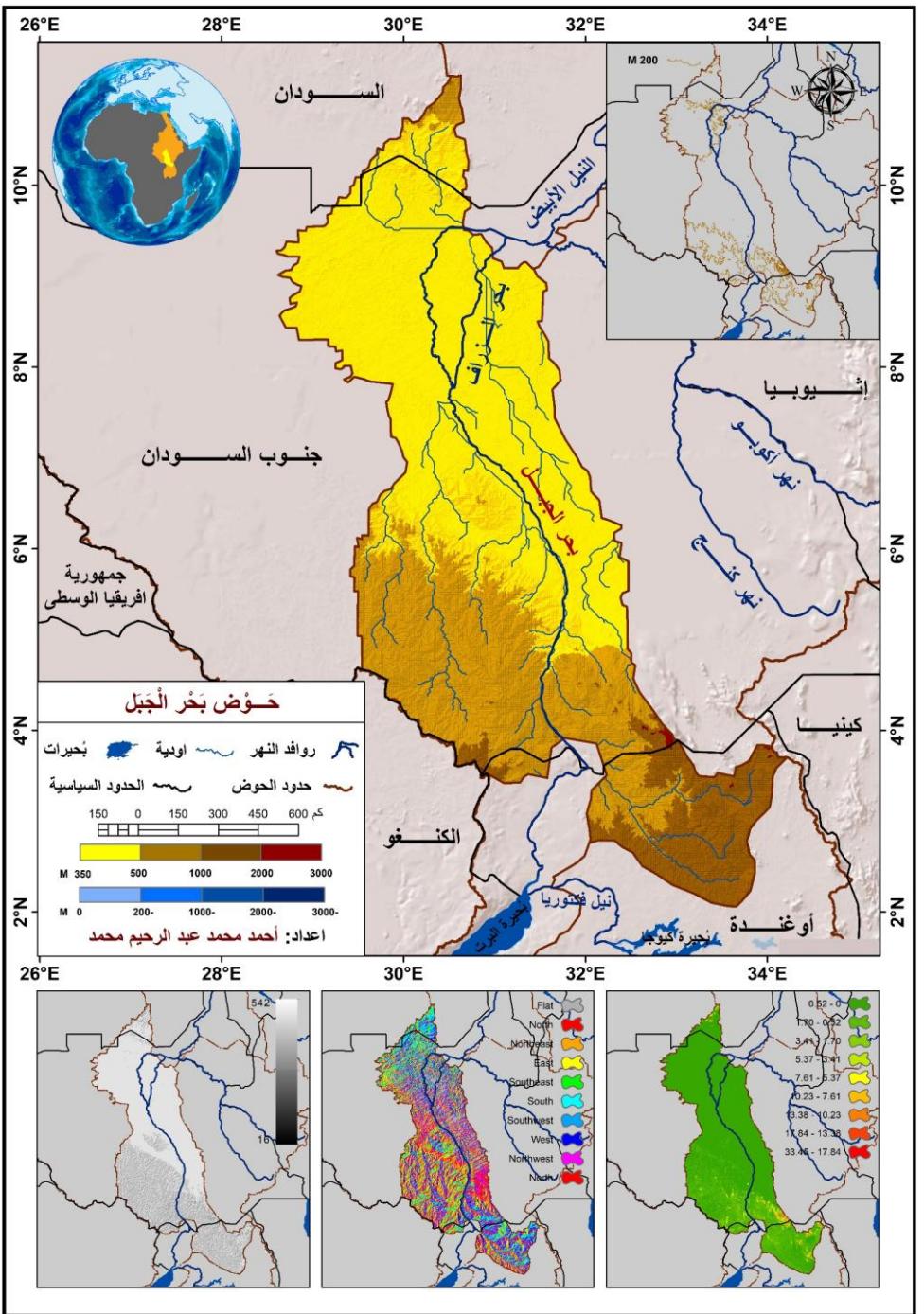
College Projects

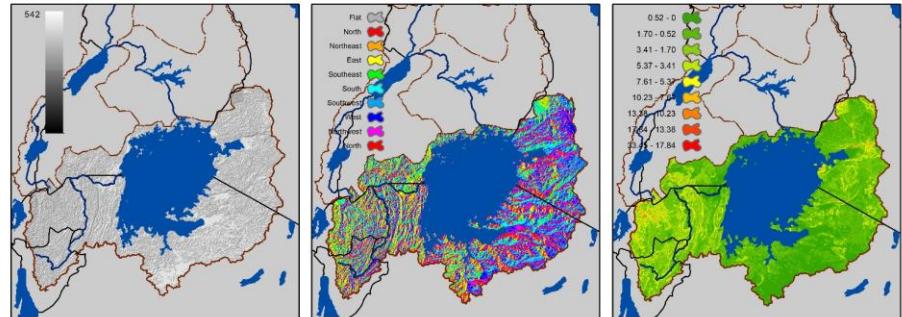
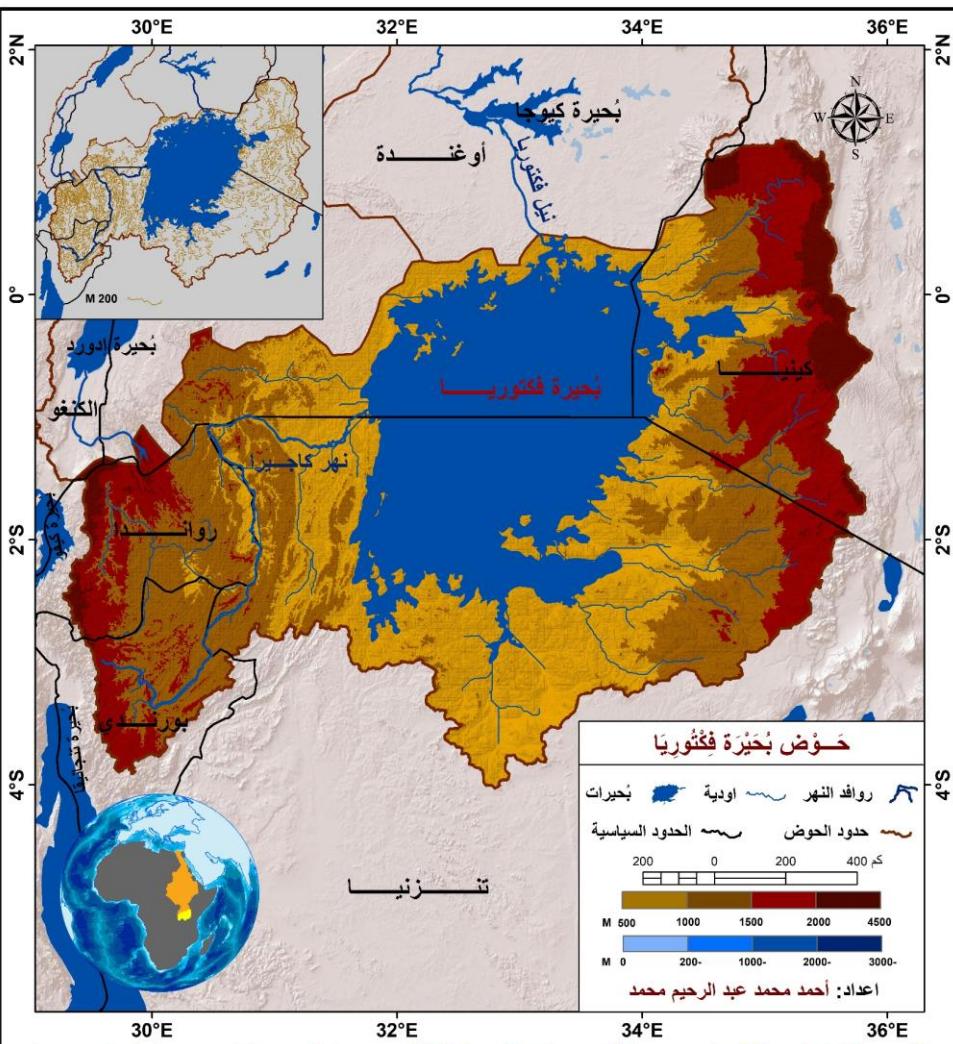
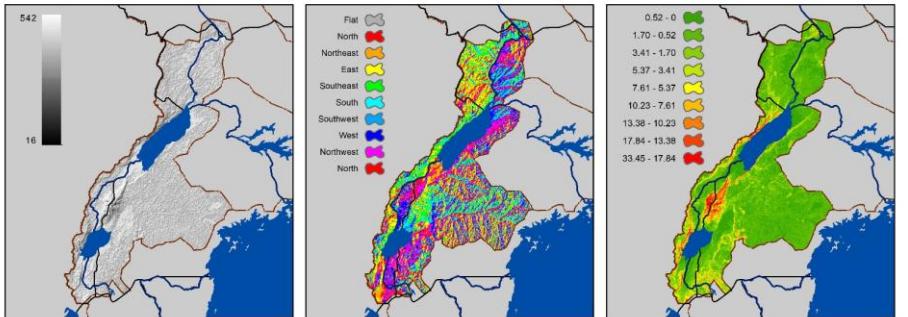
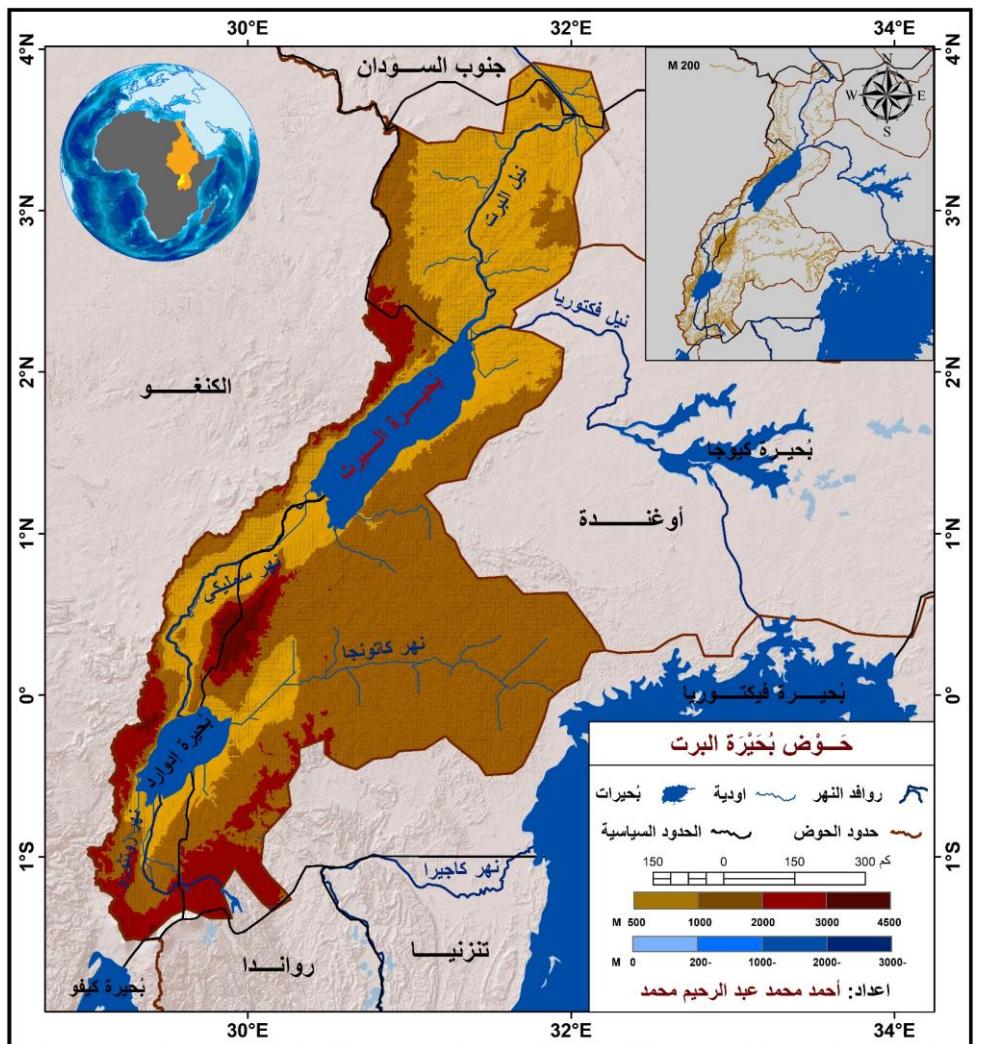


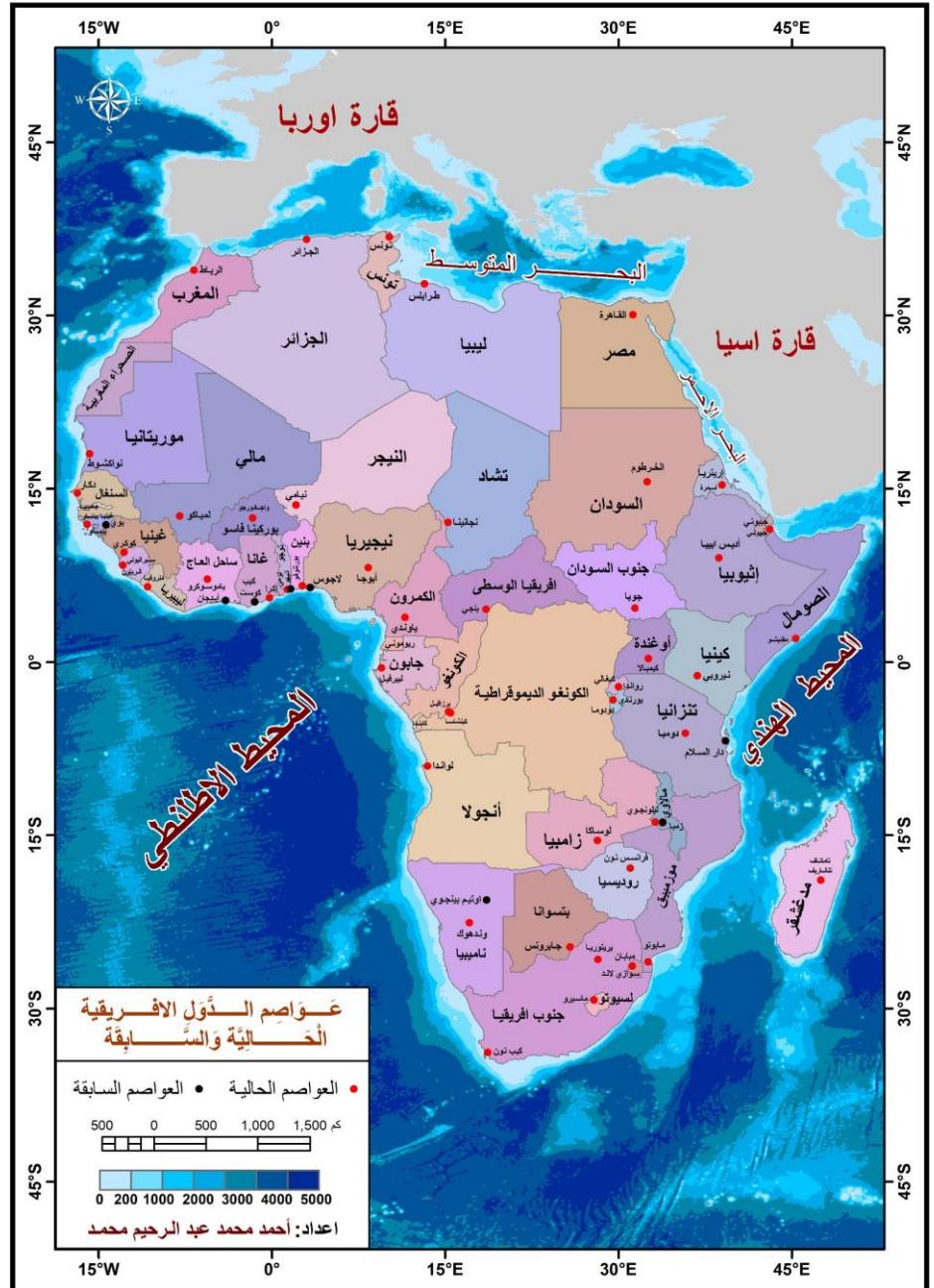
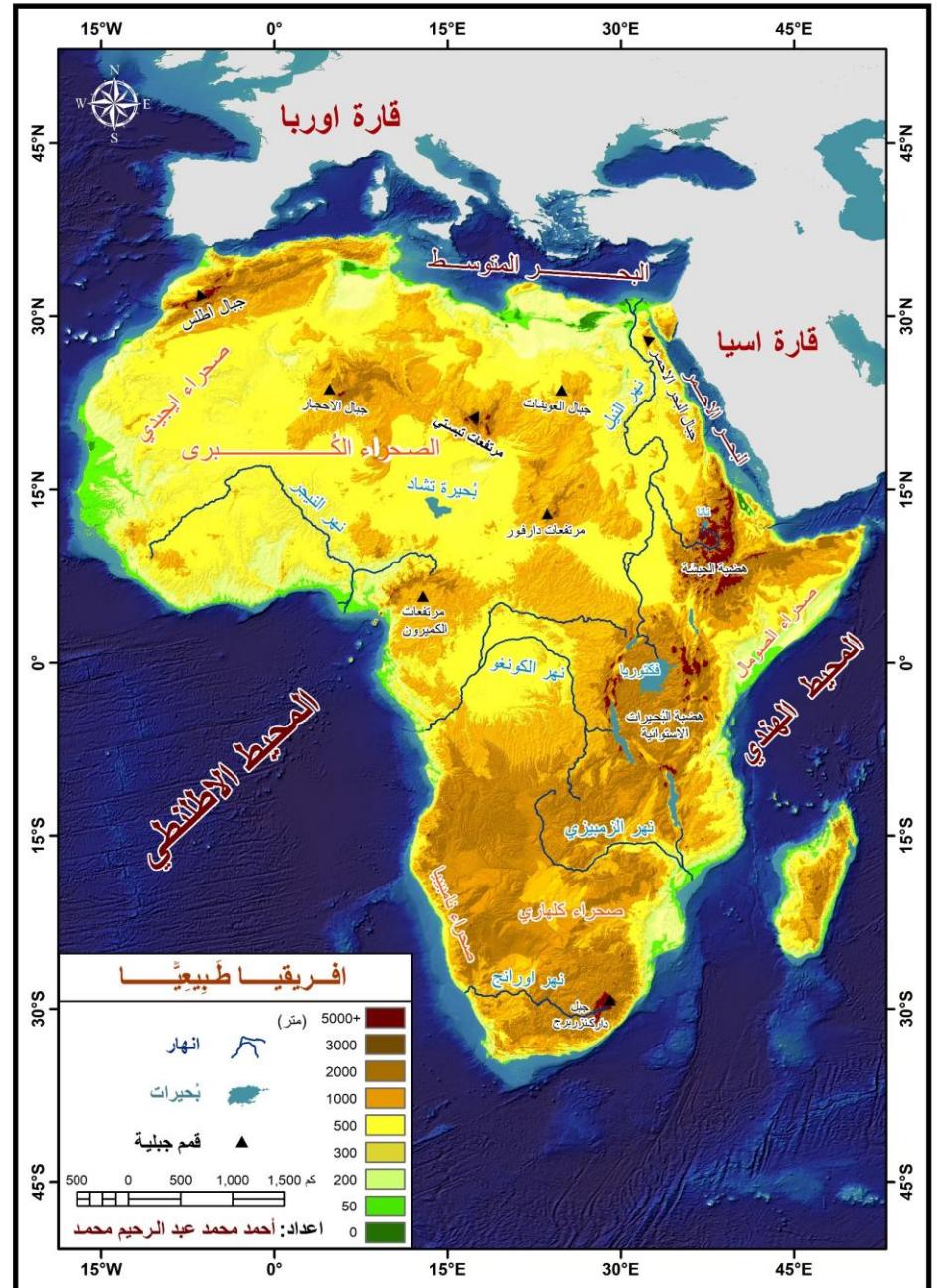


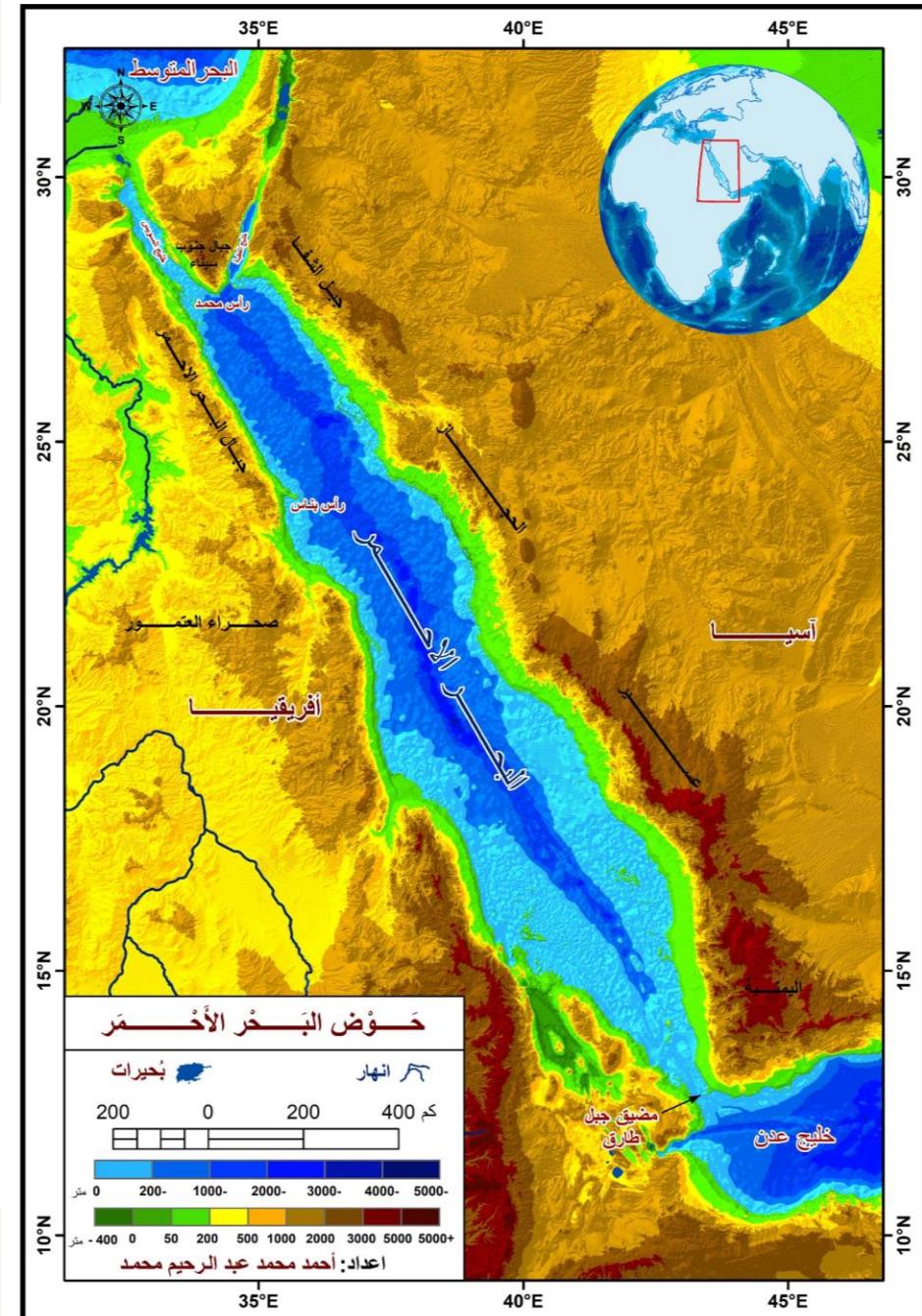
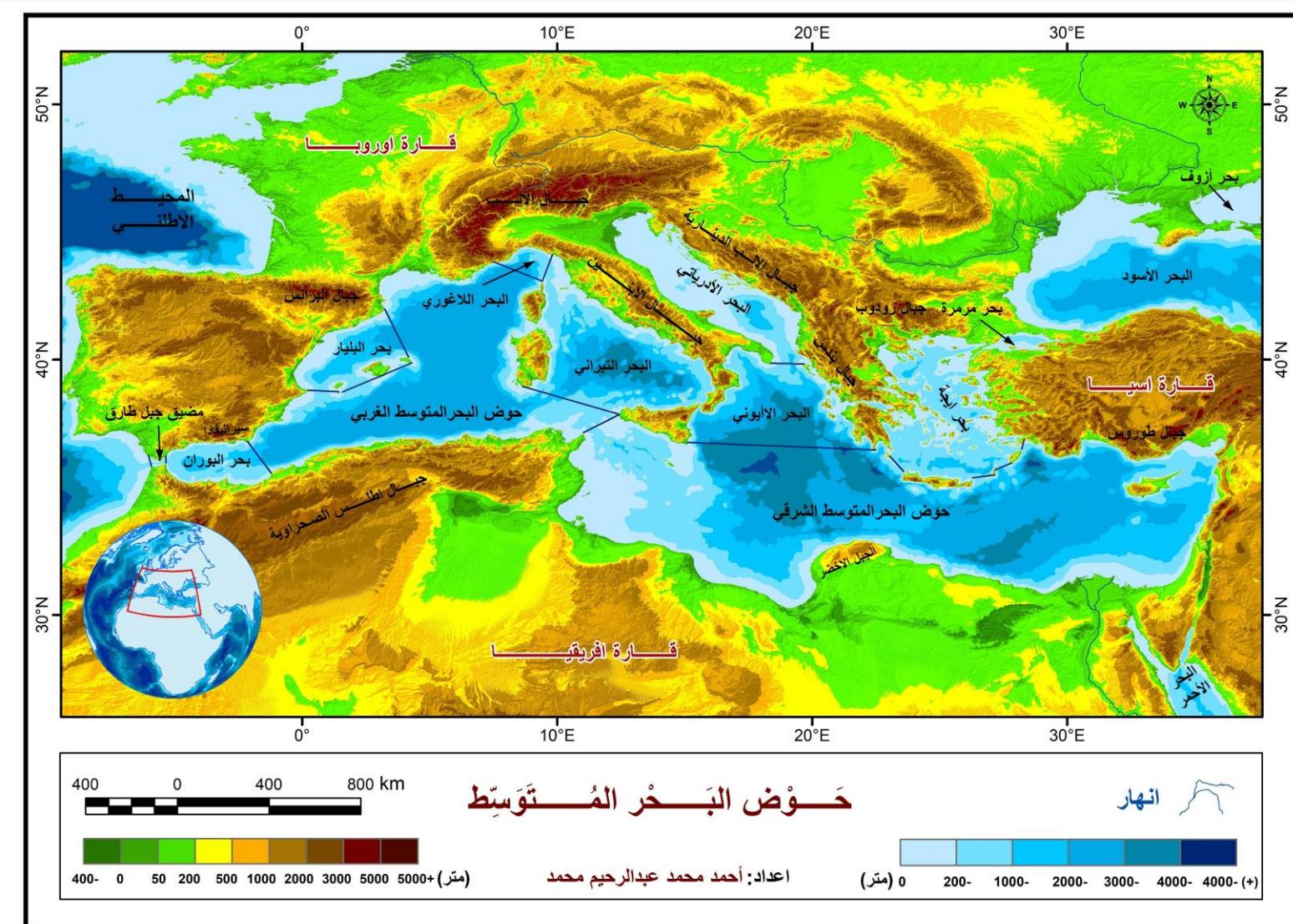


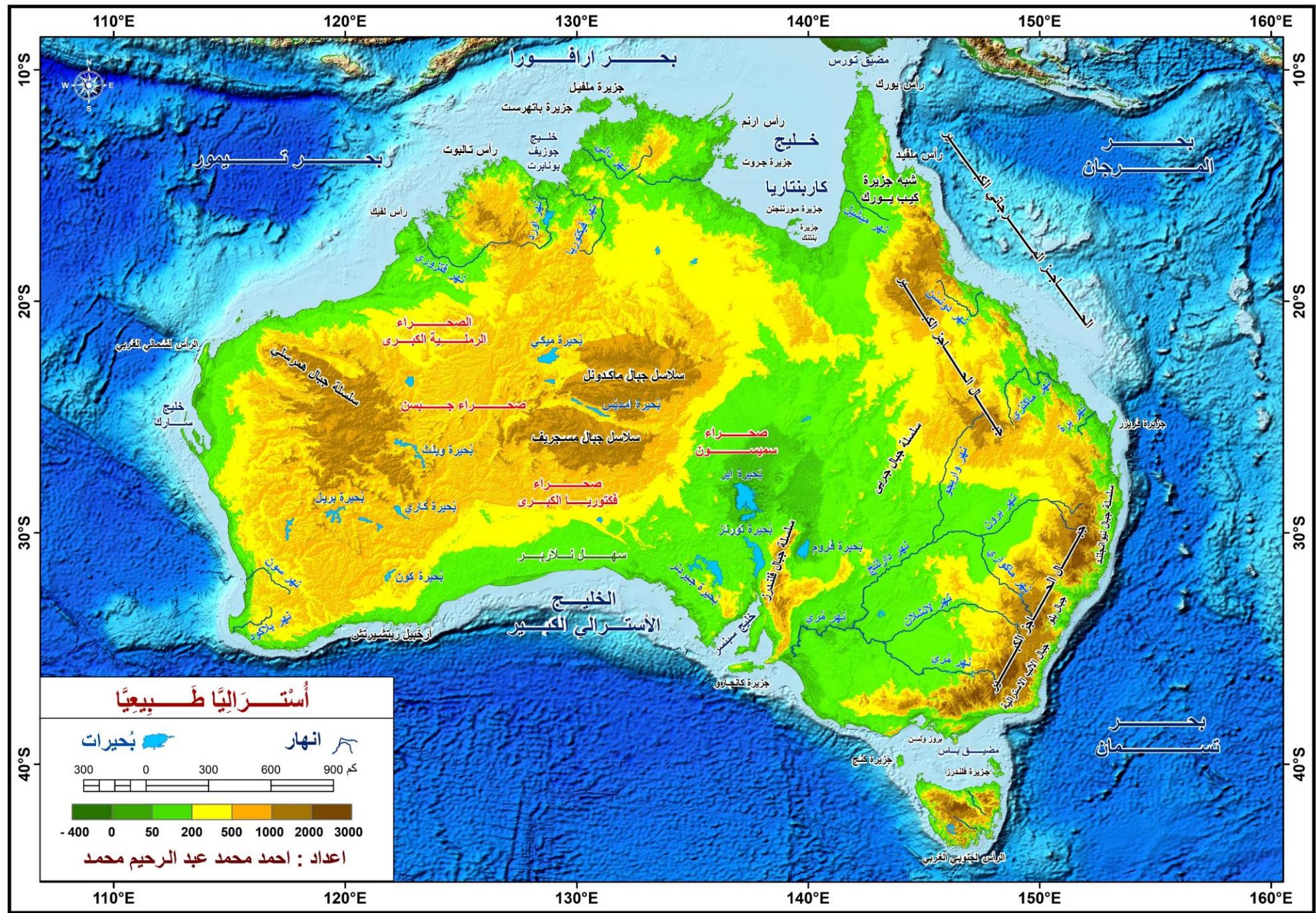


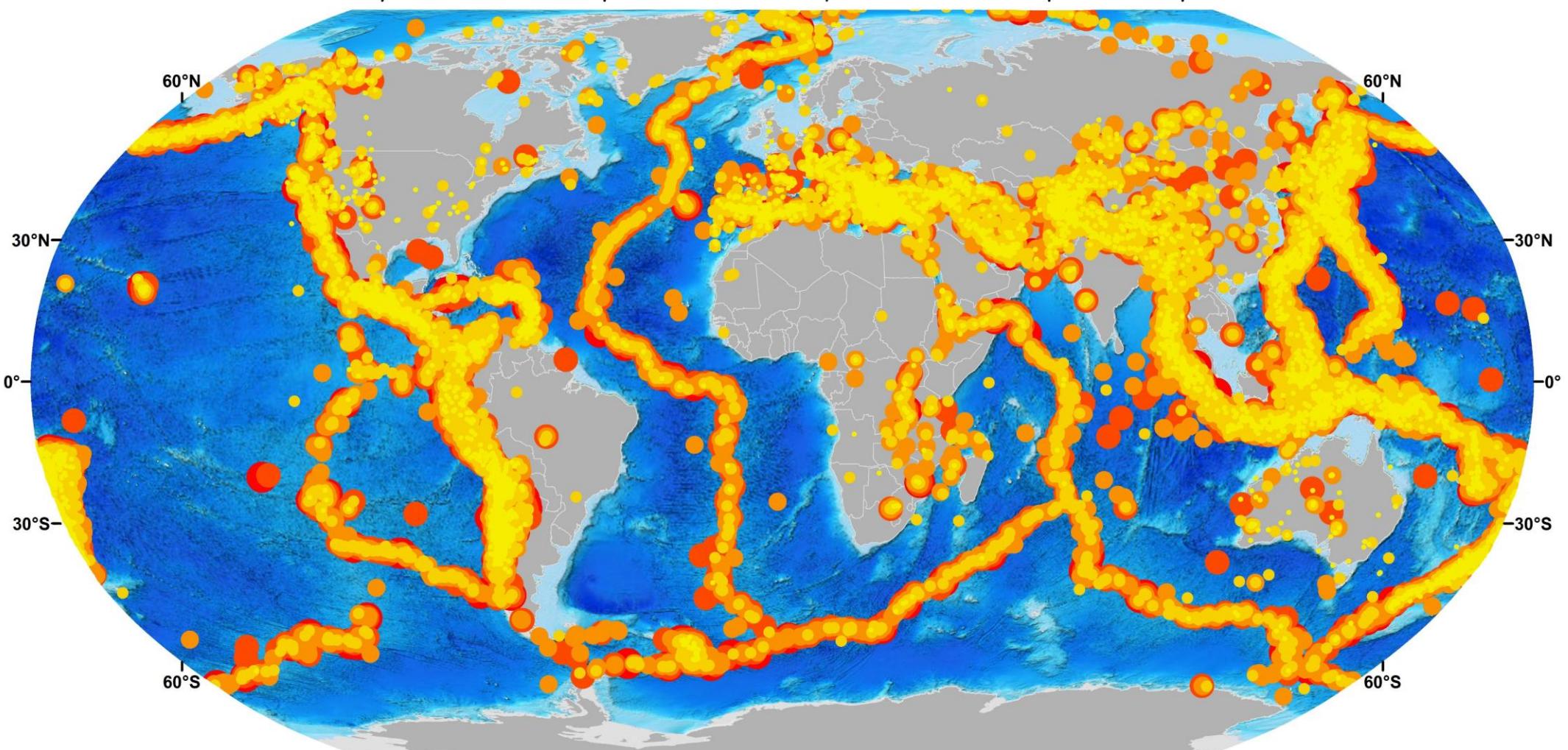












• ريختر
3.5 - 3.9

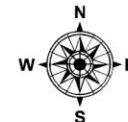
• 4 - 4.4

• 4.5 - 4.9

• 5 - 5.9

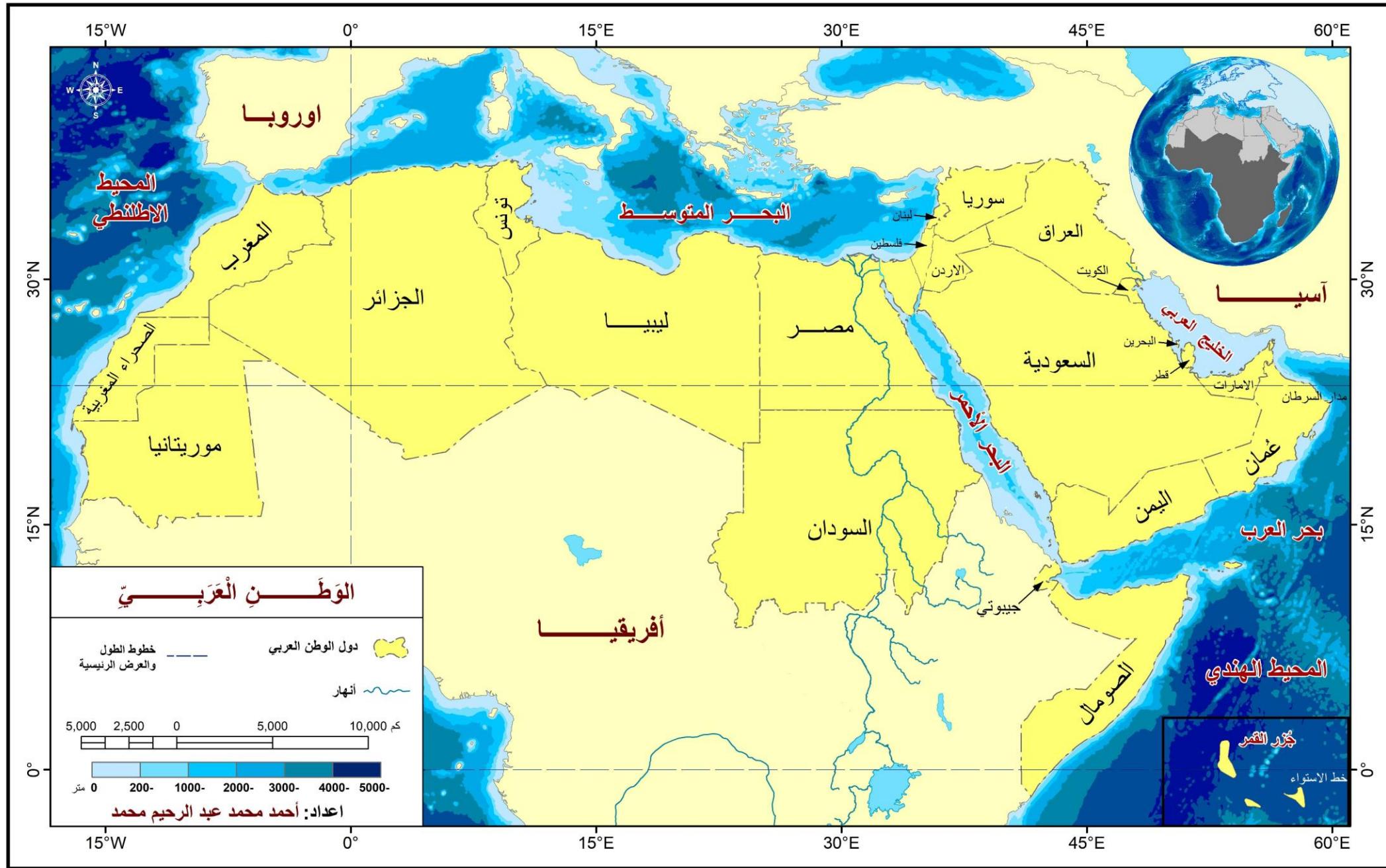
• 6 - 7.5

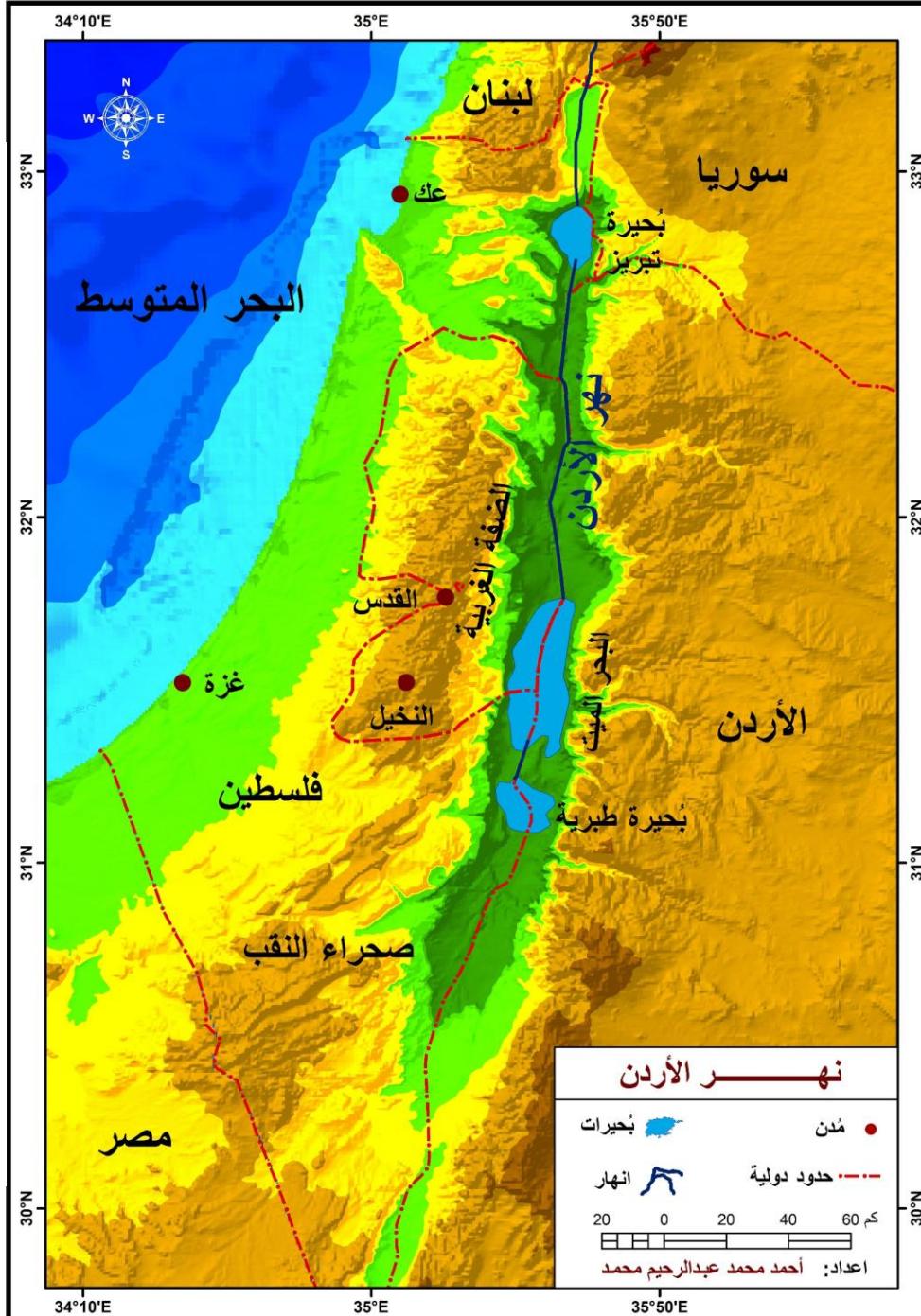
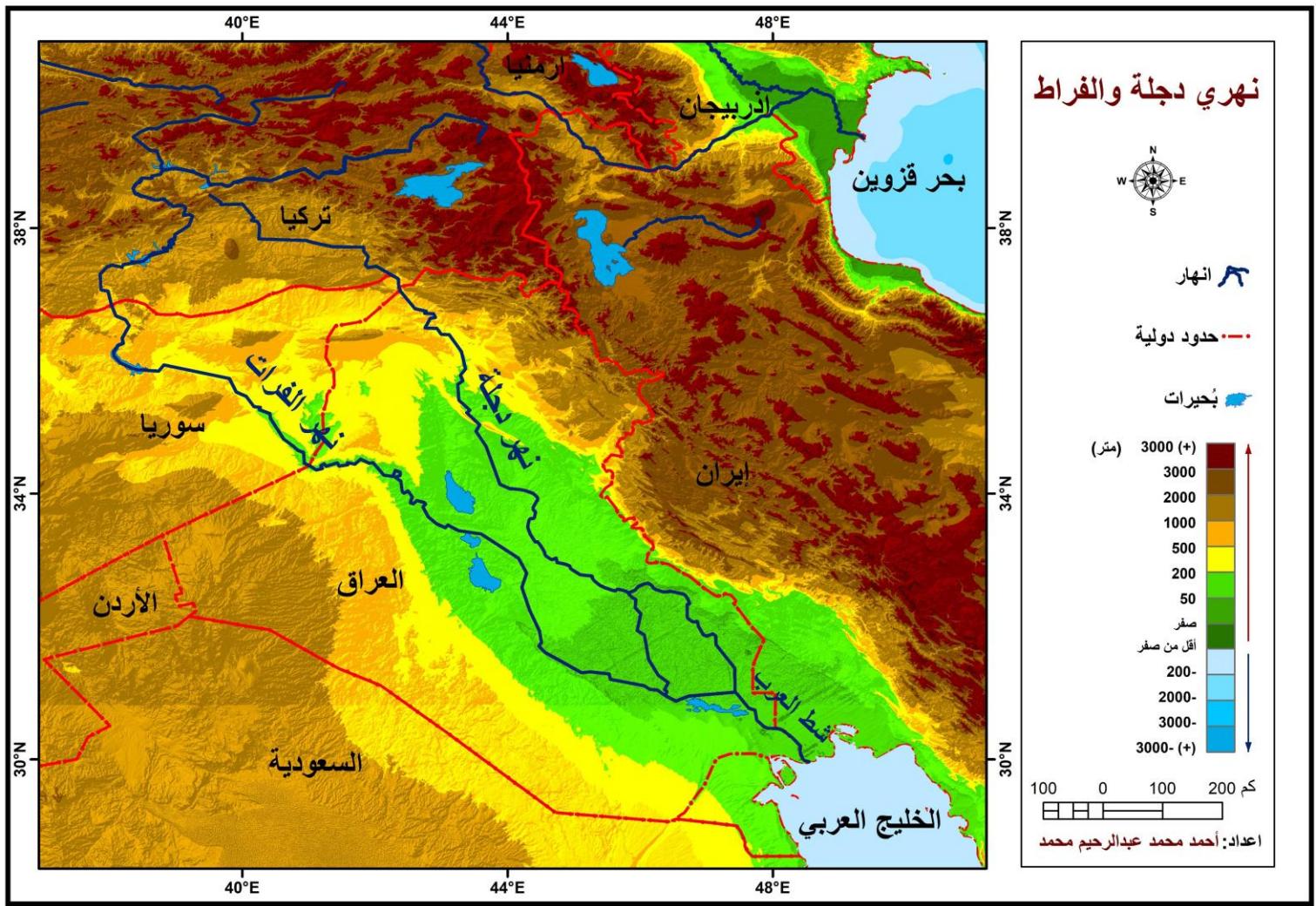
الزَّلَازِلُ عَامٌ 1997 - 2007



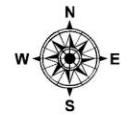
كم
2,000 0 2,000 4,000

إعداد: محمد عبد الرحيم محمد





مِصْر طَبِيعَة



جِبَل

مُدَن

أَوَدِيَّة

حَدُودُ دُولِيَّة

حَدُودُ الْمَنْخَفَضَات

كَثْبَانُ رَمِيلَة

3000 (+) مِتَر

3000

2000

1000

500

200

50

صَفَر

أَقْلَى مِنْ صَفَر

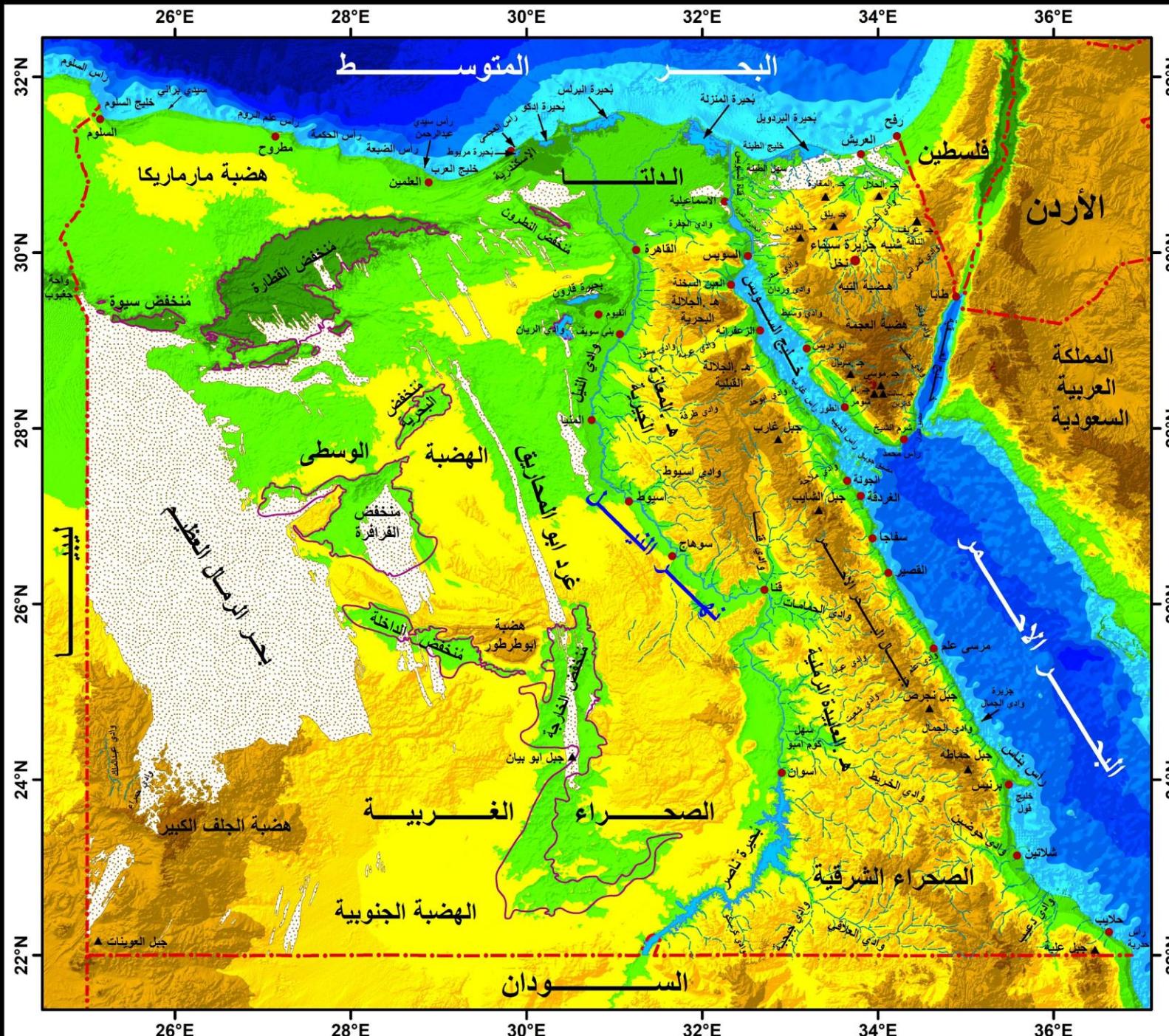
2000-

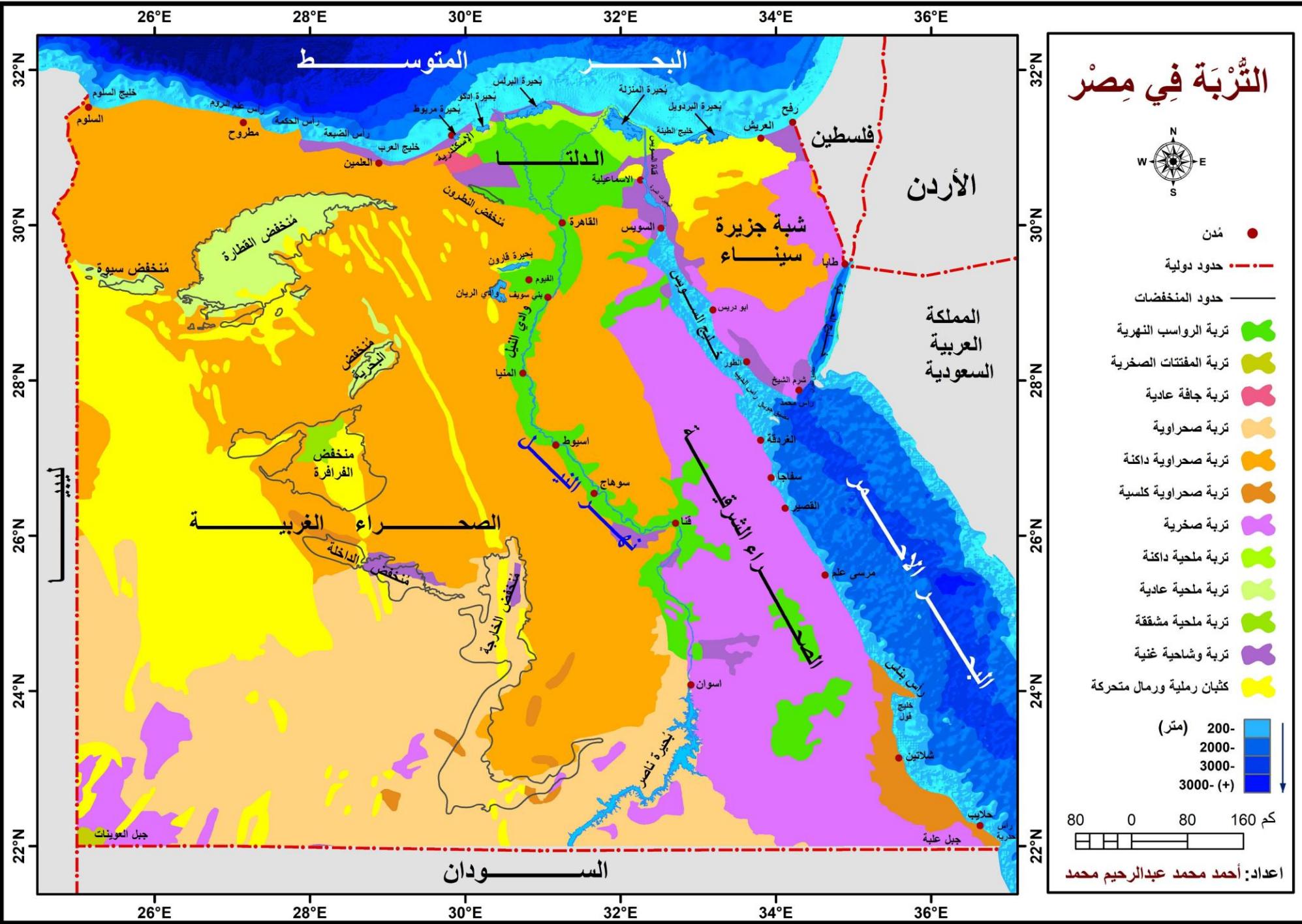
3000-

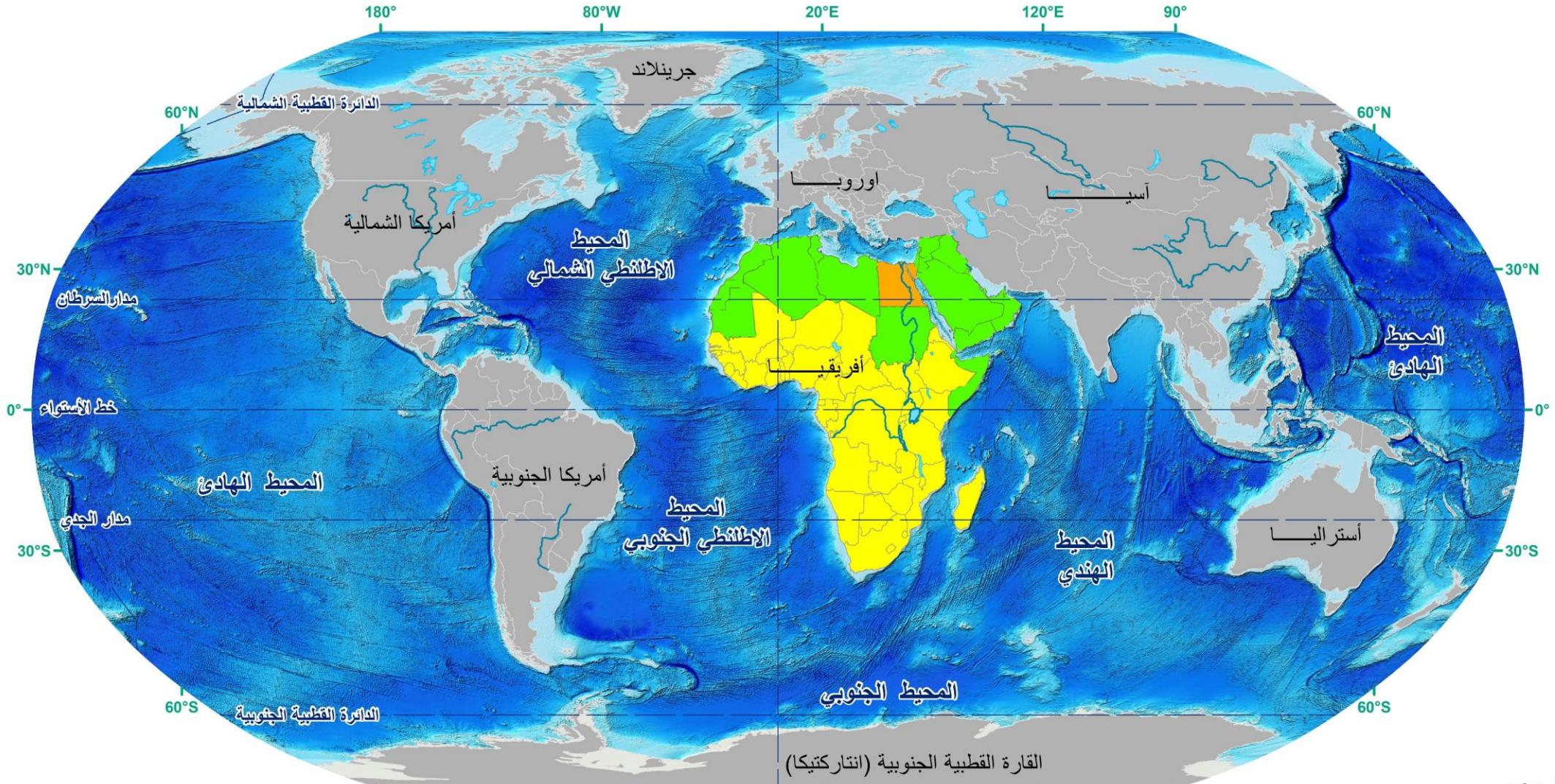
3000 (-)

80 0 80 160 كِم

اَعْدَاد: اَحْمَدُ مُحَمَّدٌ عَبْدالرَّحِيمٌ مُحَمَّد



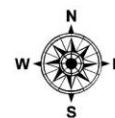




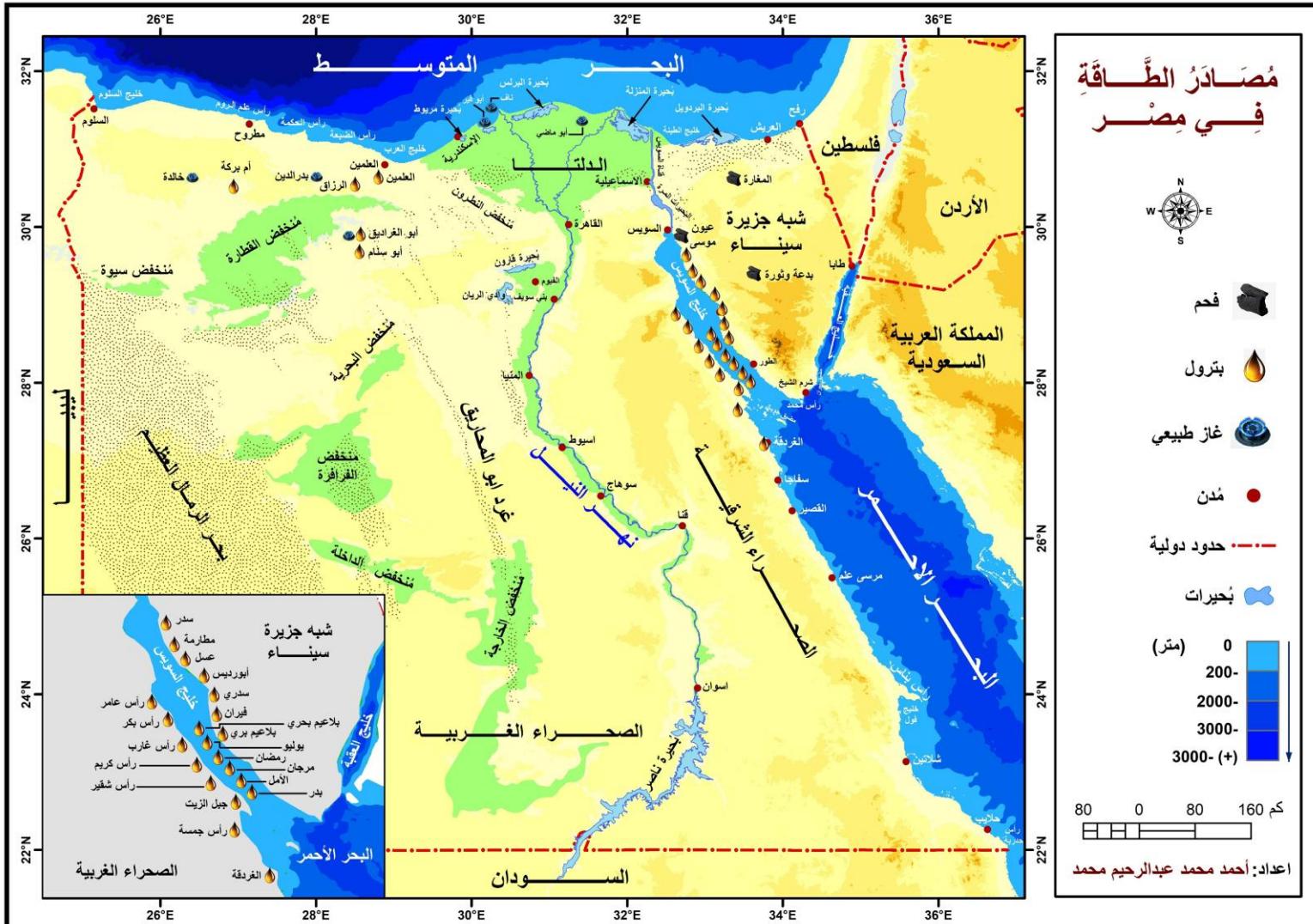
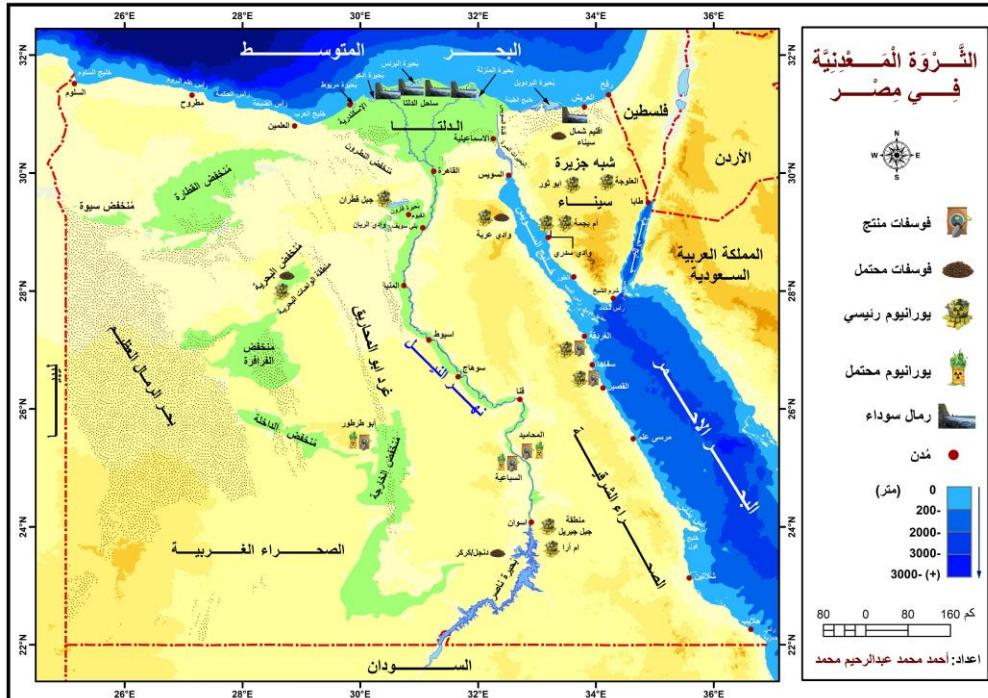
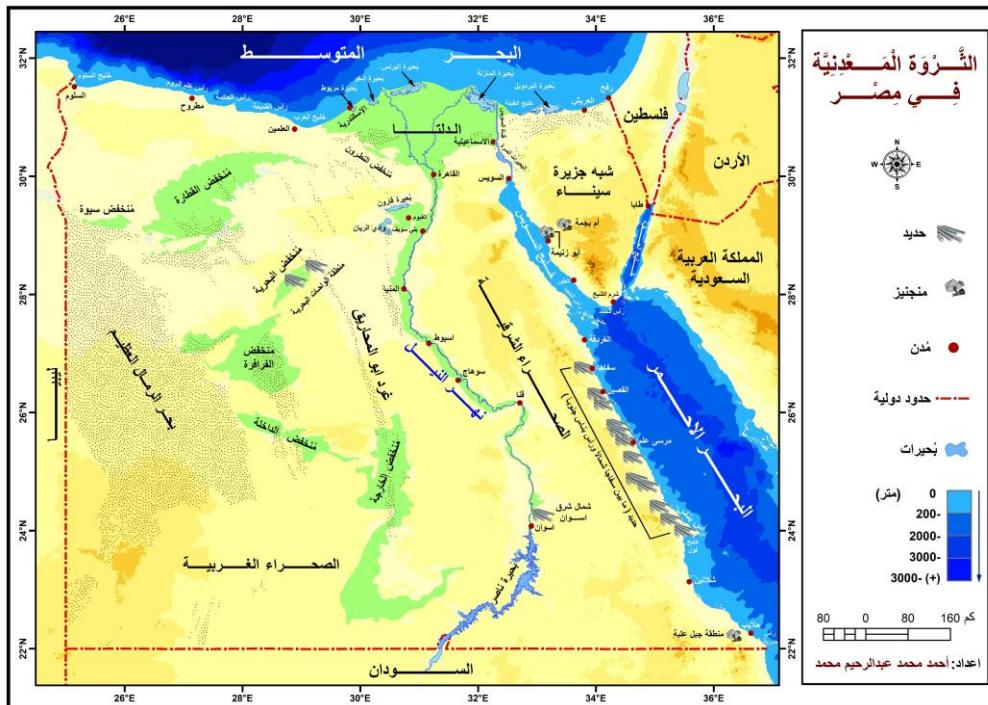
أنهار
 خطوط الطول والعرض الرئيسية
 2,000 0 2,000 4,000 كم

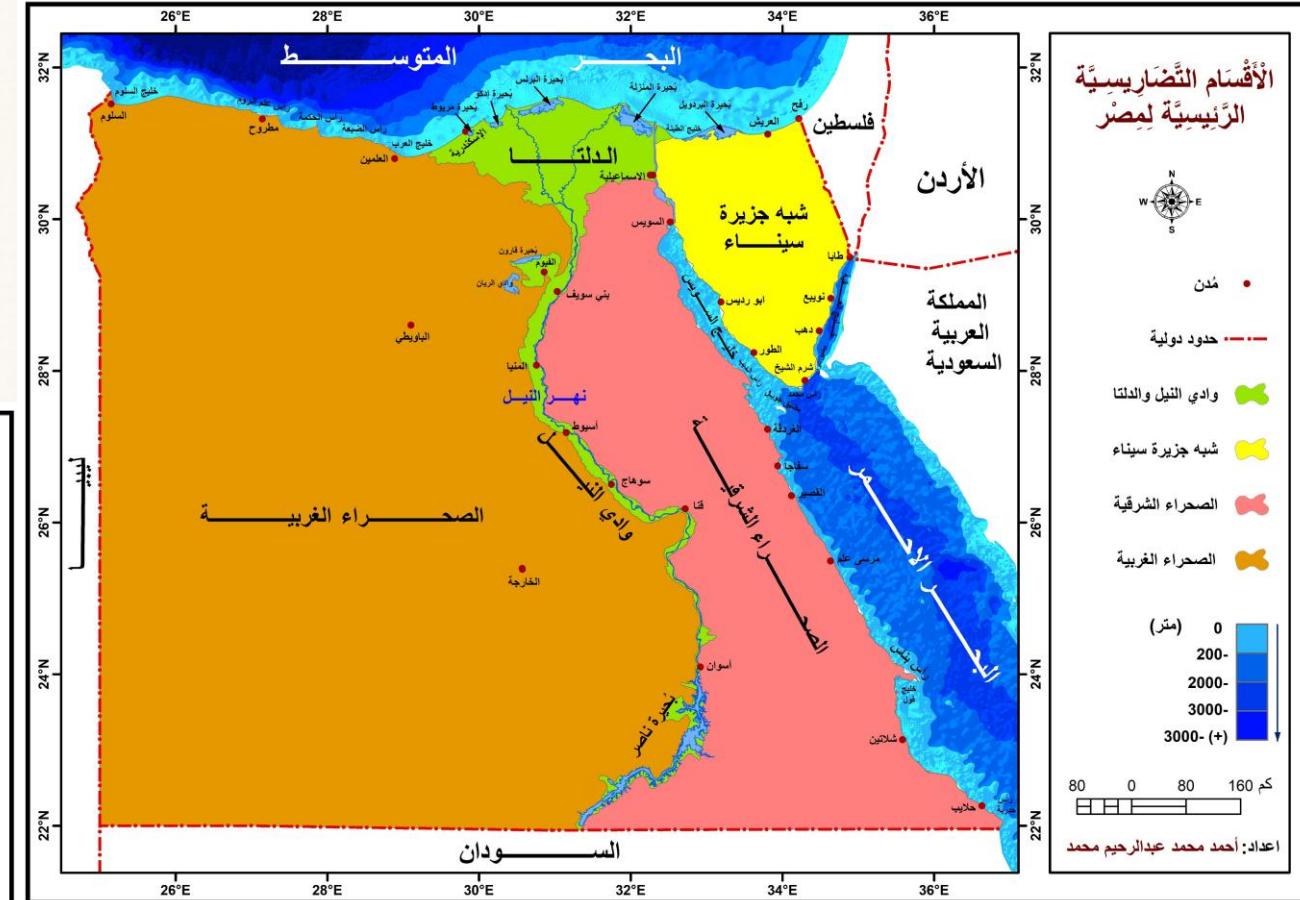
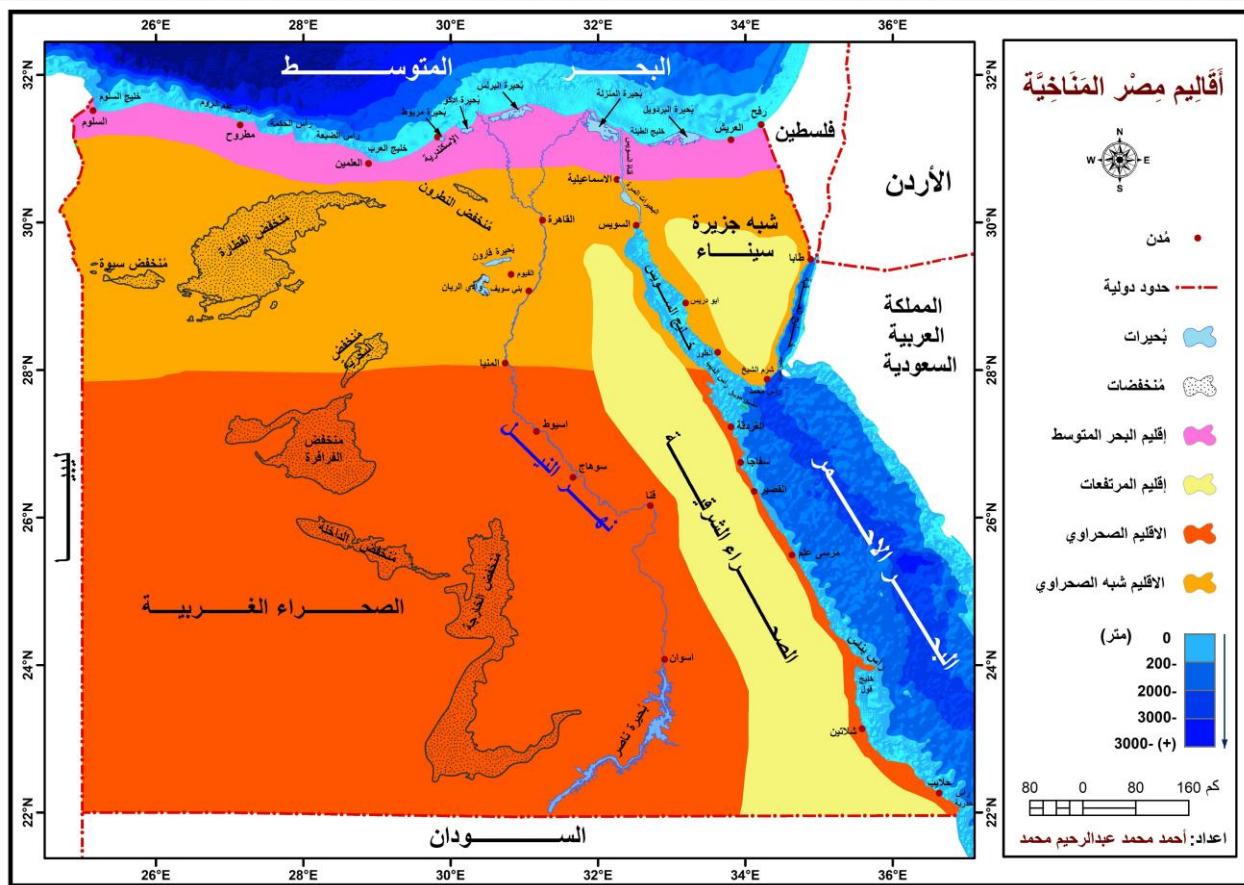
إعداد: أحمد محمد عبدالرحيم محمد

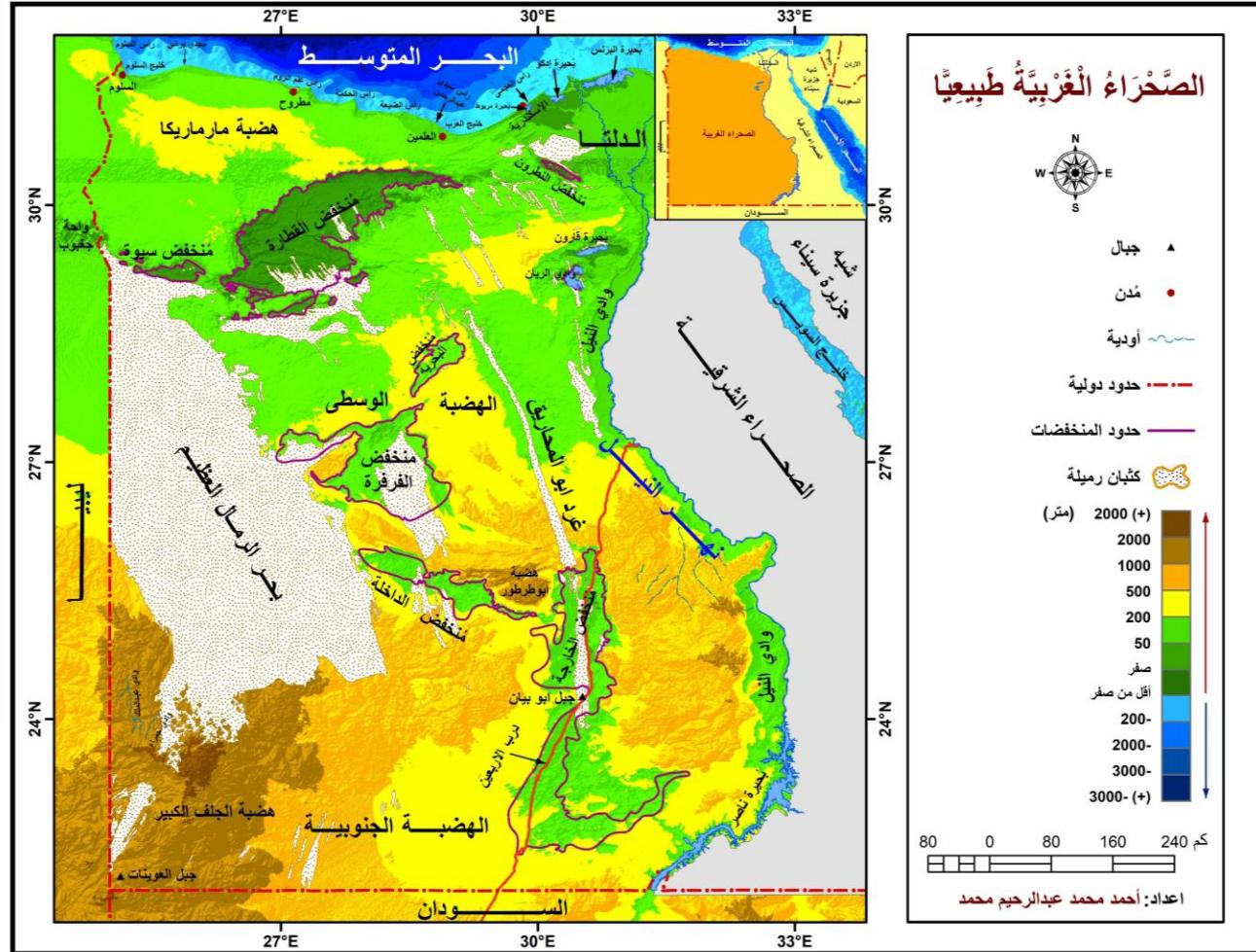
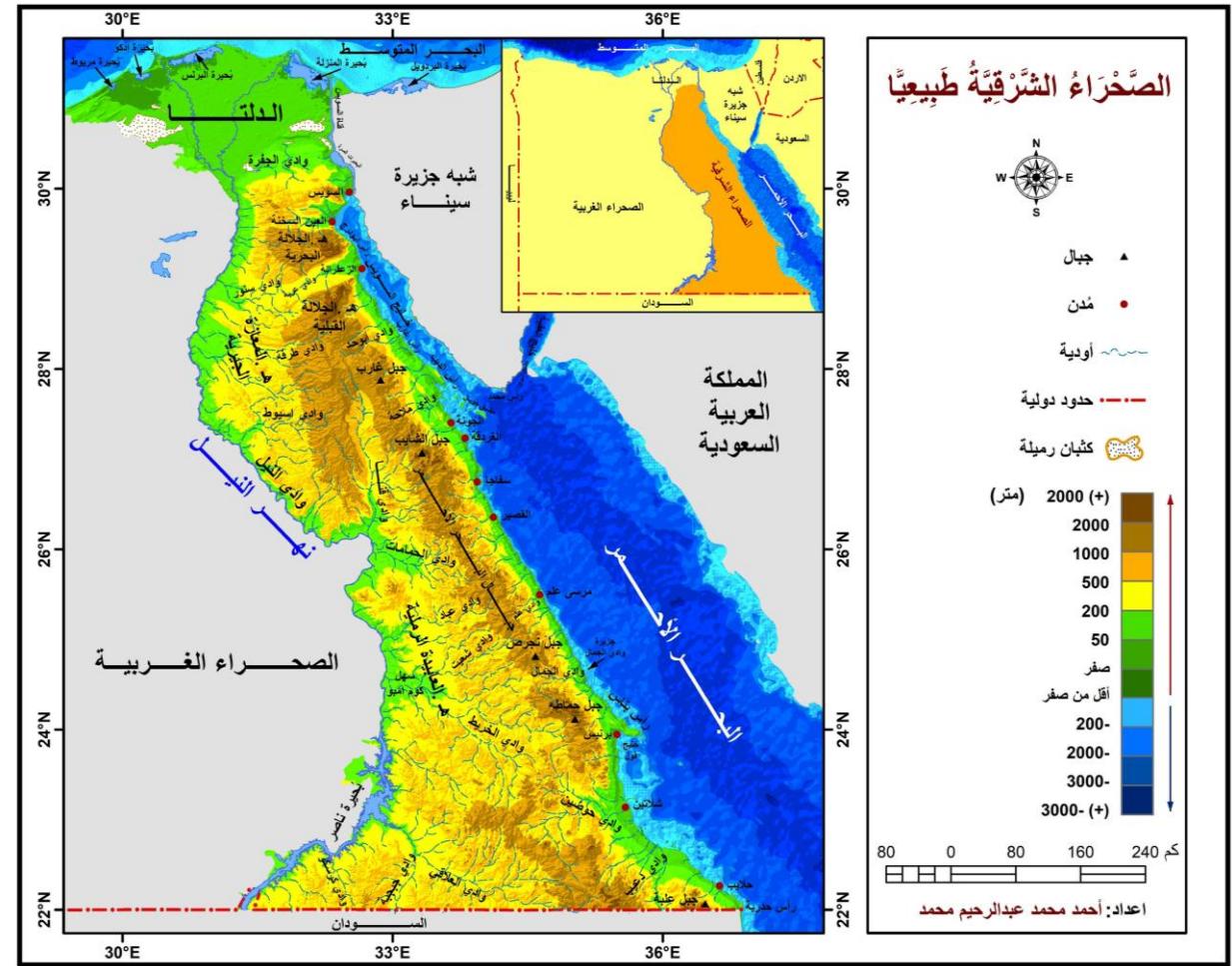
موقع مصر الجغرافي من العالم وأفريقيا والوطن العربي

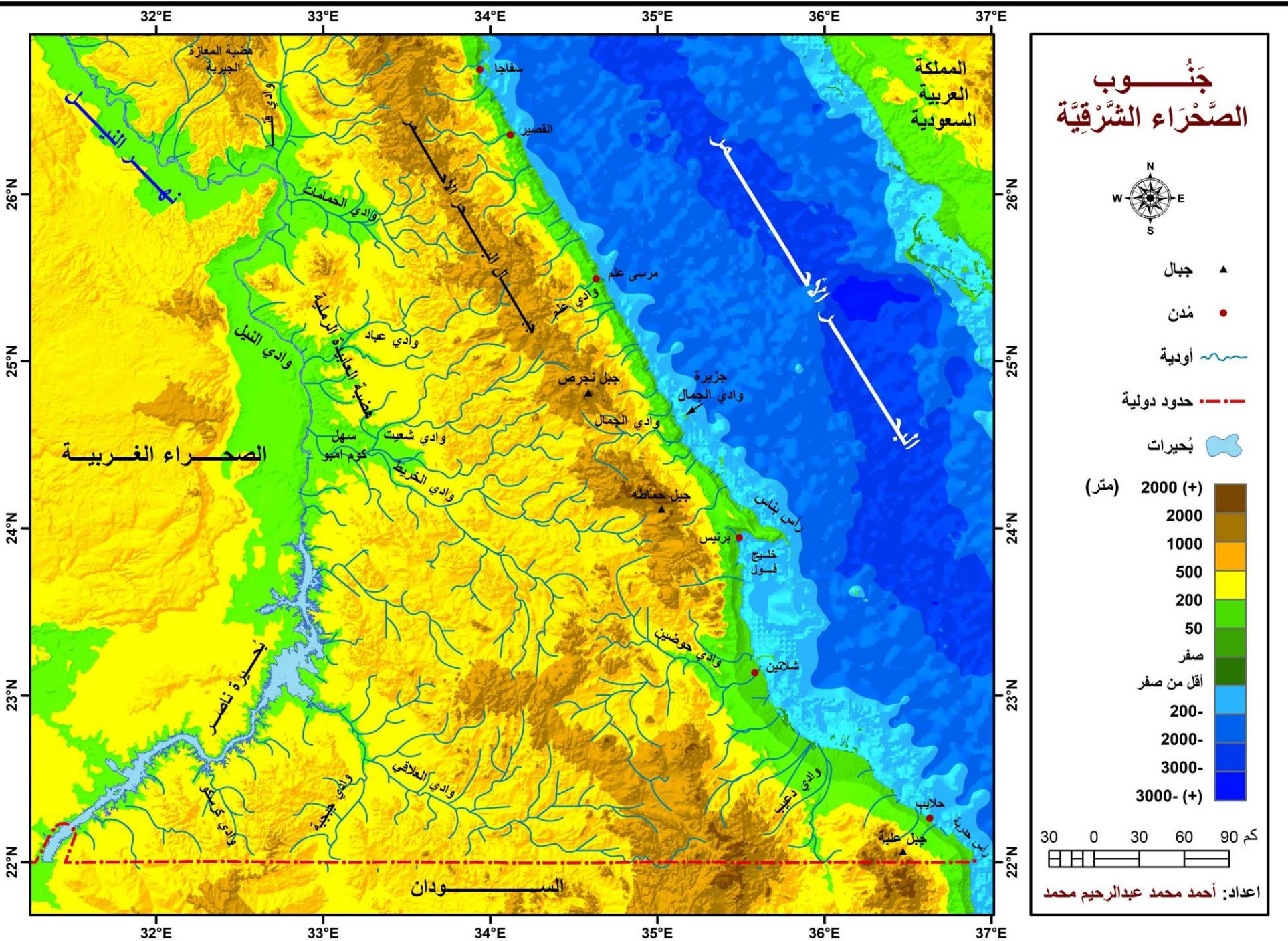


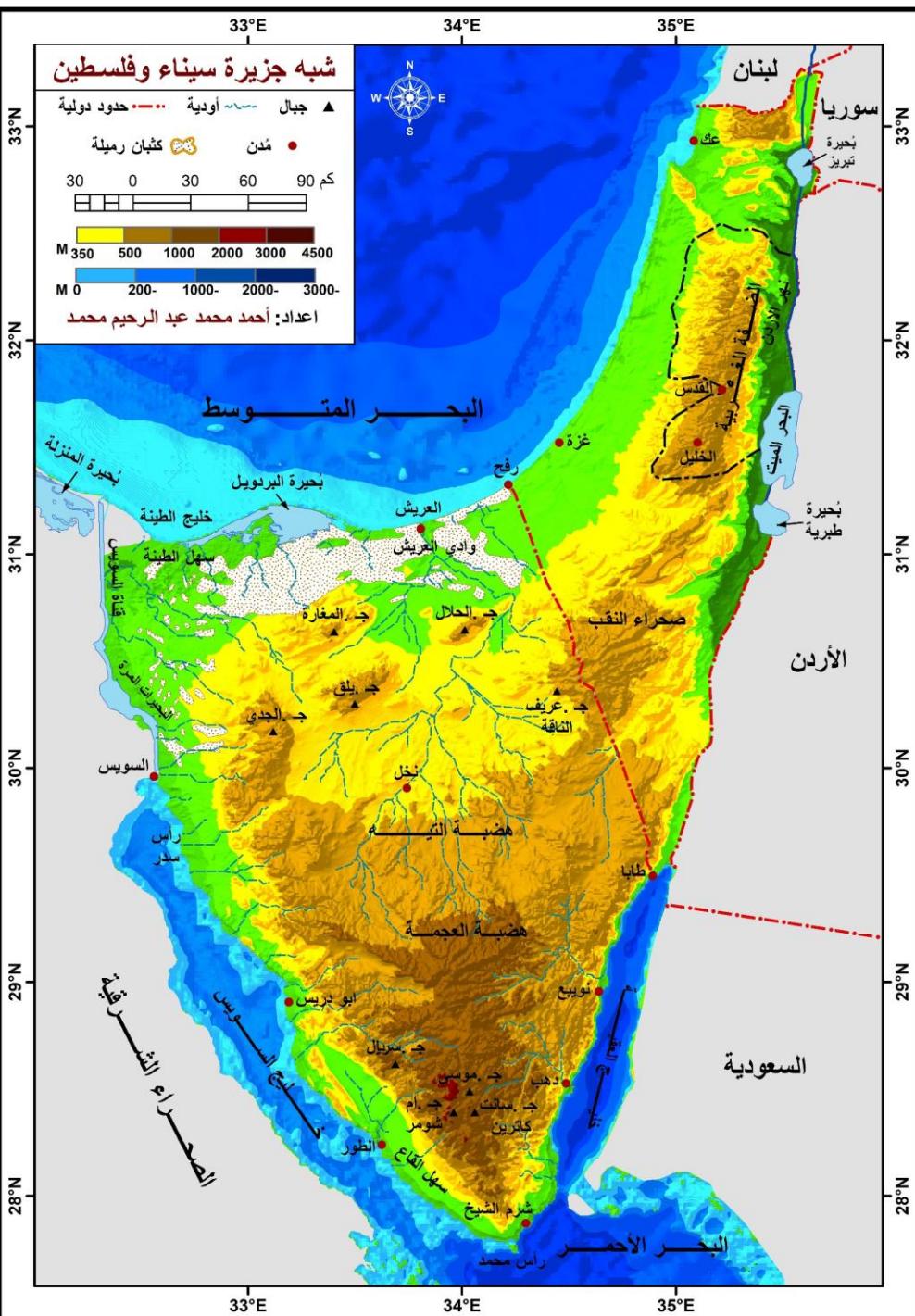
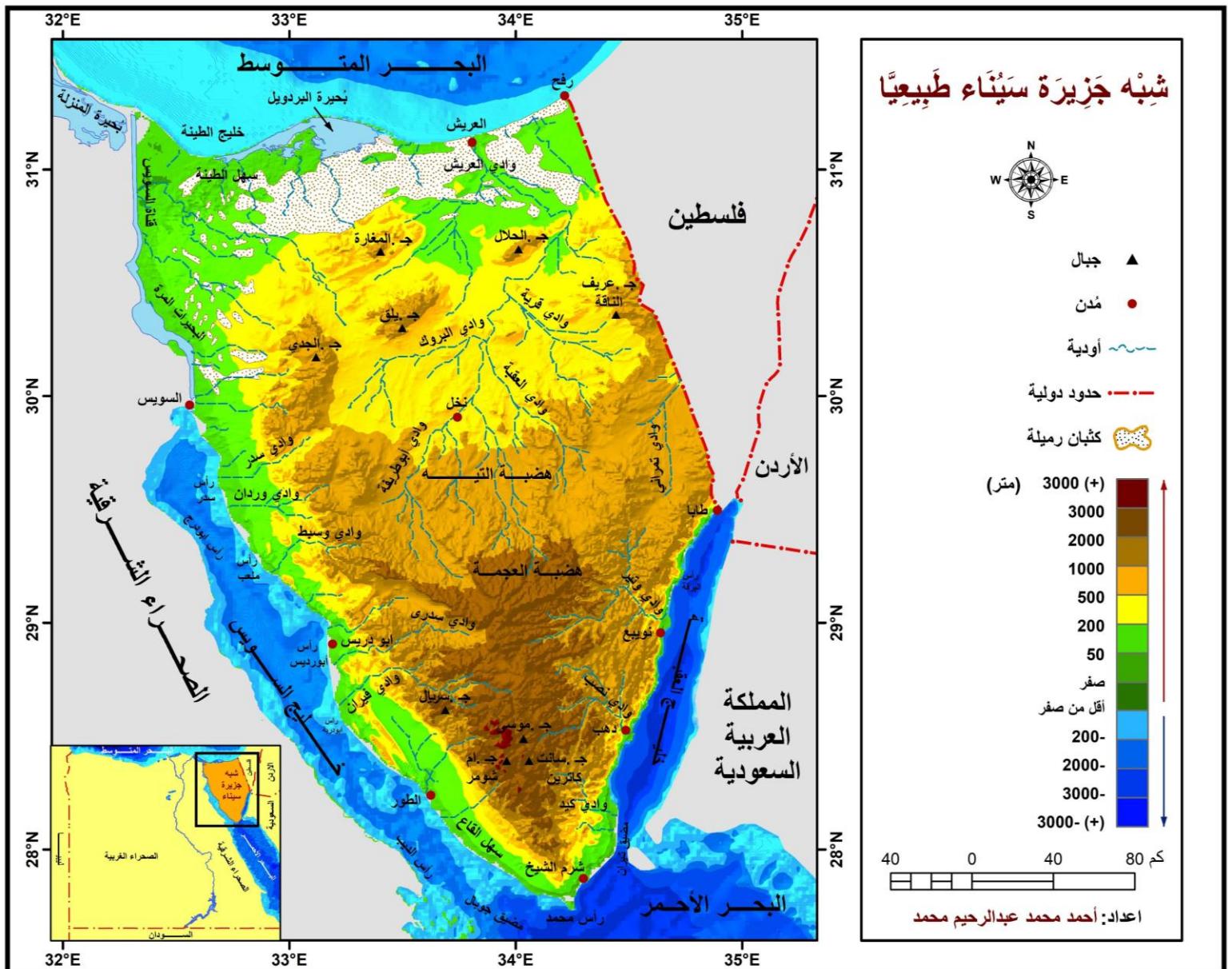
- مصر
- الوطن العربي
- افريقيا
- دول العالم
- بحيرات

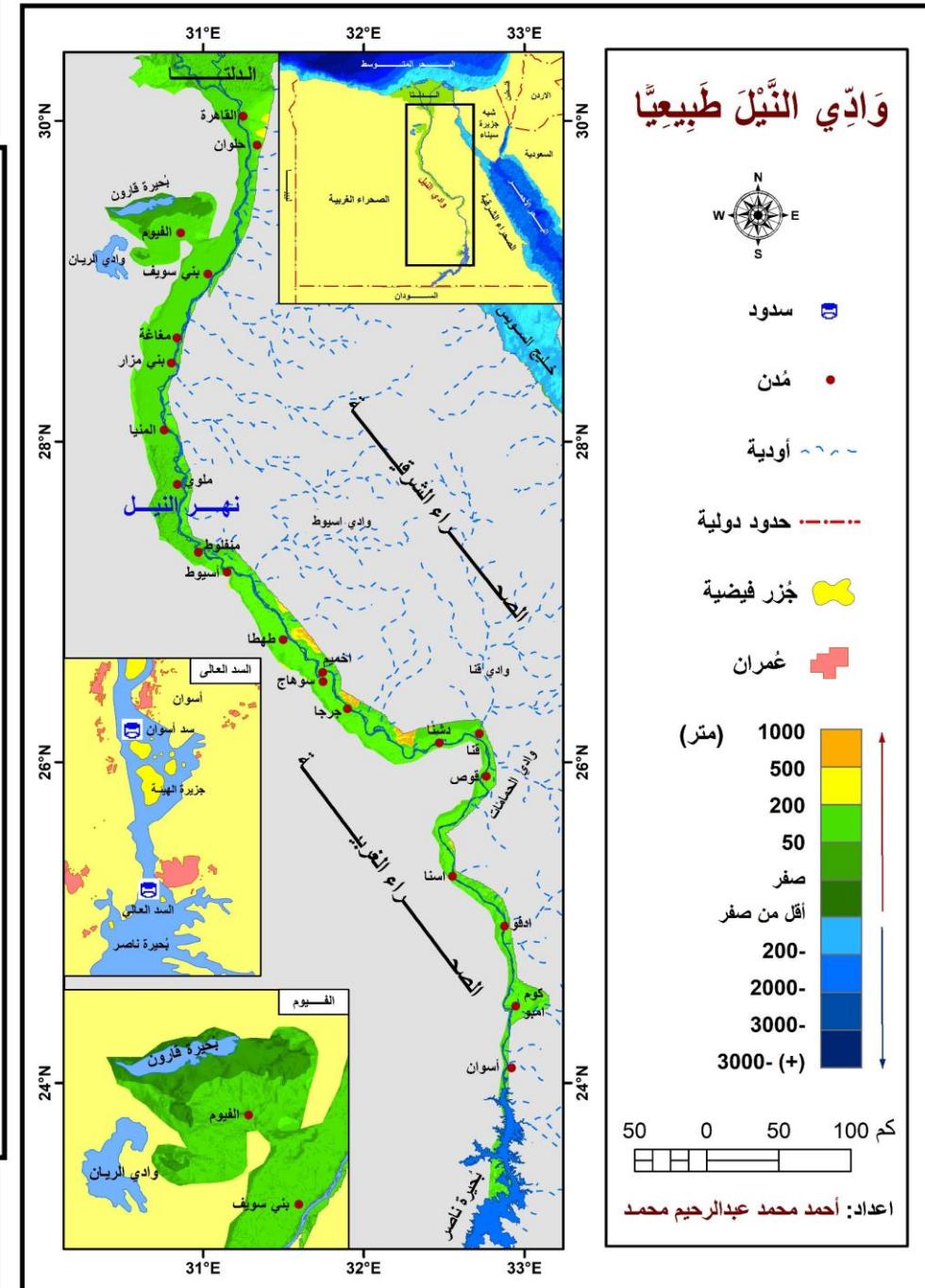
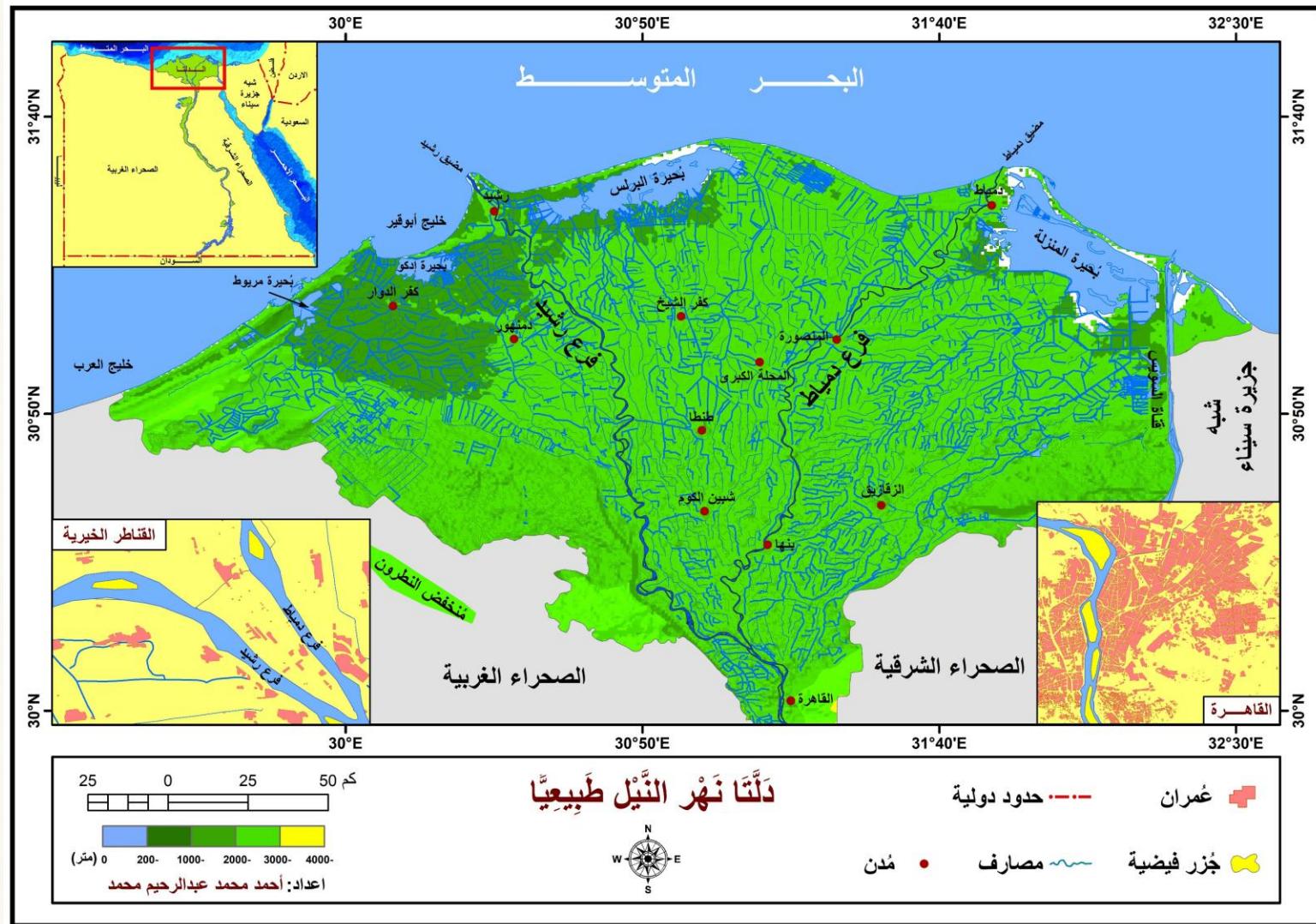














Certificates



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has successfully completed the training course of

Spatial Analysis with ArcGIS Pro

3 days of training

Completed on December 10,2023

Mohamed Magdy

Mohamed Magdy
Esri Certified Instructor

Hassem Hemeda

Hassem Hemeda
CEO

22EG72-14



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has successfully completed the training course of

Imagery Analysis in ArcGIS Pro

2 days of training

Completed on November 23,2023

Mohamed Magdy

Mohamed Magdy
Esri Certified Instructor

Hassem Hemeda

Hassem Hemeda
CEO

22EG72-14



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has successfully completed the training course of

Working with ArcGIS Network Analyst

2 days of training

Completed on November 16,2023

Elham Soudy

Elham Soudy
Esri Certified Instructor

Hassem Hemeda

Hassem Hemeda
CEO

22EG72-14



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has successfully completed the training course of

Managing Geospatial Data in ArcGIS

2 days of training

Completed on September 17,2023

Ahmed Fathy

Ahmed Fathy
Esri Certified Instructor

Hassem Hemeda

Hassem Hemeda
CEO

22EG72-14



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has successfully completed the training course of

Creating and Editing Data with ArcGIS Pro

2 days of training

Completed on September 24,2023

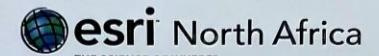
Elham Soudy

Elham Soudy
Esri Certified Instructor

Hassem Hemeda

Hassem Hemeda
CEO

22EG72-14



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has successfully completed the training course of

Creating Maps and Visualizations with ArcGIS

3 days of training

Completed on November 5,2023

Maryam .Abdel Tawab

Maryam Abdel Tawab
Esri Certified Instructor

Hassem Hemeda

Hassem Hemeda
CEO

22EG72-14



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has successfully completed the training course of

ArcGIS Pro: Essential Workflows

3 days of training

Completed on September 7, 2023

Elham Soudy

Elham Soudy

Esri Certified Instructor

Hassem Hemeda

Hassem Hemeda

CEO

22EG72-14



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has successfully completed the training course of

Introduction to GIS Using ArcGIS

2 days of training

Completed on August 27, 2023

Taghreed Mohamed

Taghreed Mohamed

Esri Certified Instructor

Hassem Hemeda

Hassem Hemeda

CEO

22EG72-14



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has successfully completed the training course of

Introducing Data Reviewer for Data Quality

2 days of training

Completed on October 22, 2023

Elham Soudy

Elham Soudy

Esri Certified Instructor

Hassem Hemeda

Hassem Hemeda

CEO

22EG72-14



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has successfully completed the training course of

Field Data Collection and Management Using ArcGIS

2 days of training

Completed on October 15, 2023

Elham Soudy

Elham Soudy

Esri Certified Instructor

Hassem Hemeda

Hassem Hemeda

CEO

22EG72-14



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has successfully completed the training course of

Introduction to Esri Story Maps

1 day of training

Completed on October 26, 2023

Maryam Abdell Tawab

Maryam Abdell Tawab

Esri Certified Instructor

Hassem Hemeda

Hassem Hemeda

CEO

22EG72-14



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has attended the MOOC

Going Places with Spatial Analysis

6 weeks

Completed on March 6, 2024

Jack Dangermond

Jack Dangermond

President



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has attended the MOOC
Cartography.

6 weeks
Completed on May 20, 2024



Jack Dangermond
President

Esri Training | esri.com/training



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has attended the web course
Python for Everyone

4 hours and 15 minutes
Completed on February 6, 2024



Jack Dangermond, President



Esri Training | esri.com/training



hereby recognizes that

Ahmed Mohamed Abdel Rahim

has attended the web course
Python Scripting: Modifying Layer Properties

1 hour and 50 minutes
Completed on February 14, 2024



Jack Dangermond, President



Esri Training | esri.com/training



Certificate of Completion

ITI Platform Mahara-Tech Certifies that

Ahmed Mohamed

Has successfully completed course

Python Programming Basics

- Course duration: 1 hour 38 minutes
- Completion date: March 14, 2024
- Verification code: VpDGPDKmc9





Python code

This is a Python code that converts each row in an Excel sheet into a Word file

From this to this

The image shows a development environment with several windows open:

- Code Editor:** A Python script named `word_automation.py` is displayed. The code uses `pandas` to read an Excel file and `docx` to create Word documents from each row of the Excel data.
- Terminal:** The terminal window shows the command `python word_automation.py` being run, and the output indicates the script is executing successfully.
- Excel Sheet:** An Excel spreadsheet titled "data - Excel" is shown, containing a table of data with columns: name, تاريخ (Date), رقم (ID), and place (الإسكندرية). The data consists of 14 rows of placeholder information.
- File Explorer:** A file explorer window titled "OUTPUT" shows a folder structure. It contains subfolders for "Pictures", "Videos", and "Ahmed Mohamed". Under "This PC", there are subfolders for "Local Disk (C:)" and "Google Drive (G:)". Inside the "Local Disk (C:)" folder, there are Microsoft Word documents for each row of the Excel sheet, with names like "المنيا", "الجيزة", "القليوبية", "القاهرة", "الإسكندرية", "اسوان", "المنيا", "الجيزة", "القليوبية", "الإسكندرية", "اسوان", "القاهرة", and "المنيا".

E: > interview > code > TSGSA_DATA_APP 10.1.1 > TSGSA_DATA_APP.py > ...

```
def ADD_POINTS():
    wb.save()

#####
###YOUSSEF###YOUSSEF###YOUSSEF###YOUSSEF###YOUSSEF###YOUSSEF###YOUSSEF###YOUSSEF###YOUSSEF###YOUSSEF###

def EXCEL_CREATE():
    path3 = path.get()
    print(path3)
    os.chdir(path3)
    filesx = os.listdir(path3)

    # opening the source excel file
    filename = "Data and Points.xlsx"
    wb1 = openpyxl.load_workbook(filename)
    ws1 = wb1.worksheets[0]

    # opening the destination excel file
    filename1 = "output.xlsx"
    wb2 = openpyxl.load_workbook(filename1)
    ws2 = wb2.active

    # calculate total number of rows and
    # columns in source excel file
    mr = ws1.max_row
    mc = ws1.max_column

    target = [[2, 2], [2, 4], [3, 2], [4, 2], [4, 4], [5, 2], [5, 4], [6, 2], [6, 4], [7, 2], [7, 4], [8, 2], [8, 4], [9, 2], [10, 2], [11, 2], [12, 2], [12, 4], [11, 4], [13, 2], [13, 4], [14, 2], [14, 4], [15, 2], [16, 2], [16, 4], [17, 2], [17, 4], [18, 2], [18, 4], [19, 2], [19, 4], [20, 2], [20, 4], [21, 2], [21, 4], [22, 2], [22, 4], [23, 2], [23, 4], [24, 2], [24, 4], [25, 2], [25, 4], [26, 2], [26, 4], [27, 2], [27, 4], [28, 2], [28, 4], [29, 2], [29, 4], [30, 2], [30, 4], [31, 2], [31, 4], [32, 2], [32, 4], [33, 2], [33, 4], [34, 2], [34, 4], [35, 2], [35, 4], [36, 2], [36, 4], [37, 2], [37, 4], [38, 2], [38, 4], [39, 2], [39, 4], [40, 2], [40, 4], [41, 2], [41, 4], [42, 2], [42, 4], [43, 2], [43, 4], [44, 2], [44, 4], [45, 2], [45, 4], [46, 2], [46, 4], [47, 2], [47, 4], [48, 2], [48, 4], [49, 2], [49, 4], [50, 2], [50, 4], [51, 2], [51, 4], [52, 2], [52, 4], [53, 2], [53, 4], [54, 2], [54, 4], [55, 2], [55, 4], [56, 2], [56, 4], [57, 2], [57, 4], [58, 2], [58, 4], [59, 2], [59, 4], [60, 2], [60, 4], [61, 2], [61, 4], [62, 2], [62, 4], [63, 2], [63, 4], [64, 2], [64, 4], [65, 2], [65, 4], [66, 2], [66, 4], [67, 2], [67, 4], [68, 2], [68, 4], [69, 2], [69, 4], [70, 2], [70, 4], [71, 2], [71, 4], [72, 2], [72, 4], [73, 2], [73, 4], [74, 2], [74, 4], [75, 2], [75, 4], [76, 2], [76, 4], [77, 2], [77, 4], [78, 2], [78, 4], [79, 2], [79, 4], [80, 2], [80, 4], [81, 2], [81, 4], [82, 2], [82, 4], [83, 2], [83, 4], [84, 2], [84, 4], [85, 2], [85, 4], [86, 2], [86, 4], [87, 2], [87, 4], [88, 2], [88, 4], [89, 2], [89, 4], [90, 2], [90, 4], [91, 2], [91, 4], [92, 2], [92, 4], [93, 2], [93, 4], [94, 2], [94, 4], [95, 2], [95, 4], [96, 2], [96, 4], [97, 2], [97, 4], [98, 2], [98, 4], [99, 2], [99, 4], [100, 2], [100, 4], [101, 2], [101, 4], [102, 2], [102, 4], [103, 2], [103, 4], [104, 2], [104, 4], [105, 2], [105, 4], [106, 2], [106, 4], [107, 2], [107, 4], [108, 2], [108, 4], [109, 2], [109, 4], [110, 2], [110, 4], [111, 2], [111, 4], [112, 2], [112, 4], [113, 2], [113, 4], [114, 2], [114, 4], [115, 2], [115, 4], [116, 2], [116, 4], [117, 2], [117, 4], [118, 2], [118, 4], [119, 2], [119, 4], [120, 2], [120, 4], [121, 2], [121, 4], [122, 2], [122, 4], [123, 2], [123, 4], [124, 2], [124, 4], [125, 2], [125, 4], [126, 2], [126, 4], [127, 2], [127, 4], [128, 2], [128, 4], [129, 2], [129, 4], [130, 2], [130, 4], [131, 2], [131, 4], [132, 2], [132, 4], [133, 2], [133, 4], [134, 2], [134, 4], [135, 2], [135, 4], [136, 2], [136, 4], [137, 2], [137, 4], #####
```

This is a Python code that converts each row in an Excel sheet into a Excel sheet

This is a Python code
that converts each row
in an Excel sheet into
a Excel sheet

From this to this

This image shows a dual-screen setup illustrating a Python script for generating QR codes from Excel data.

Left Screen (Code Editor):

- The code editor displays a Python script named `generate_qrcodes.py`.
- The script uses the `os`, `pandas`, `qrCode`, and `tkinter` libraries.
- The main function `generate_qr_codes()` reads an Excel file, extracts URLs and names, generates QR codes for each row, and saves them with the corresponding names.

Right Screen (Excel Workbook):

- An Excel sheet titled "Sheet1" contains data with columns: URL, X&Y, Name, X, and Y.
- The data consists of 15 rows, each representing a location with its coordinates and a unique identifier.
- The "Output" tab shows the generated QR codes for the first 10 rows.

Annotations:

- A large red arrow points from the text "From this to this" to the Excel sheet.
- Text in the center says: "This is a Python code that converts X, Y coordinates into a Google Maps link and then converts the link into a QR code".
- Below the text, there are 10 QR codes corresponding to the first 10 rows of data.

This is a Python code that crops images to any size you want

```
from PIL import Image
import os

left = 405
top = 0
right = 1518
bottom = 770

directory = r"E:\interview\code\crop_photos\Input"

for filename in os.listdir(directory):
    img_path = os.path.join(directory, filename)
    img = Image.open(img_path)
    img_res = img.crop(left, top, right, bottom)

    # create a new file name with "cropped_"
    new_path = os.path.join(
        r"E:\interview\code\crop_photos\Output", filename
    )
    # save the cropped image with the new file name
    img_res.save(new_path)
```

The code uses the PIL library to open images from a directory, crop them using specified coordinates, and save them with a new file name in an output directory.

The Input folder contains 10 images named AHMED-01-MO-1000 through AHMED-01-MO-1009. The Output folder contains 10 images named AHMED-01-M-O-1000 through AHMED-01-M-O-1009. A yellow arrow points from the Input folder to the Output folder, labeled "From this to this".

This image shows a screenshot of a development environment, likely Microsoft Visual Studio Code, illustrating a Python script for converting Word documents to PDFs.

The main area displays a Python file named `wordtopdf.py`:

```
1 import os
2 from docx2pdf import convert
3
4 # Directory containing Word files to convert
5 word_directory = r"E:\interview\code\wordtopdf\Output"
6
7 # List all files in the directory
8 word_files = os.listdir(word_directory)
9
10 # Convert each Word file to PDF
11 for word_file in word_files:
12     try:
13         # Construct the full path to the Word file
14         word_file_path = os.path.join(word_directory, word_file)
15
16         # Convert the Word file to PDF
17         convert(word_file_path)
18         print(f"Converted {word_file} to PDF successfully.")
19     except Exception as e:
20         print(f"Conversion of {word_file} failed: {e}")
21
```

A large yellow callout box highlights the text:

**This is a Python code
that converts Word
files to PDF**

The terminal output shows the conversion process:

```
Converted 17_1707_1_00000106147.docx to PDF successfully.
100% [██████████] 1/1 [00:00<00:00, 1.34it/s]
Converted 02_0206_0_00000004305.docx to PDF successfully.
100% [██████████] 1/1 [00:00<00:00, 1.37it/s]
Converted 21_2105_0_0000146553.docx to PDF successfully.
100% [██████████] 1/1 [00:00<00:00, 1.28it/s]
Converted 21_2105_0_0000146579.docx to PDF successfully.
PS C:\Users\Ahmed Mohamed>
```

The right side of the image shows a file explorer window titled "Output" displaying converted files:

Name	Date modified	Type
الجبرة	٢٠٢٤/٣/١٥ ص	Microsoft Word Document
الدقهلية	٢٠٢٤/٣/١٥ ص	Microsoft Word Document
Ahmed Mohamed		
Local Disk (C:)		
New Volume (E:)		
New Volume (F:)		
الفيوم	٢٠٢٤/٣/١٥ ص	Microsoft Word Document
القاهرة	٢٠٢٤/٣/١٥ ص	Microsoft Word Document
القلوبية	٢٠٢٤/٣/١٥ ص	Microsoft Word Document
المنوفية	٢٠٢٤/٣/١٥ ص	Microsoft Word Document

A red arrow points from the original code to the converted files, with the text "From this to this" written vertically between them.

This is a Python code that converts JPG to PDF

```
from PIL import Image
from reportlab.lib.pagesizes import letter
from reportlab.pdfgen import canvas
import os

def convert_jpeg_to_pdf(input_folder, output_folder):
    # Check if the output folder exists, if not, create it
    if not os.path.exists(output_folder):
        os.makedirs(output_folder)

    # List all files in the input folder
    files = os.listdir(input_folder)

    # Iterate through each file
    for file in files:
        # Check if the file is a JPEG image
        if file.lower().endswith(".jpg") or file.lower().endswith(".jpeg"):
            # Open the image
            img = Image.open(os.path.join(input_folder, file))

            # Create a PDF file with the same name as the image
            pdf_path = os.path.join(output_folder, os.path.splitext(file)[0] + ".pdf")

            # Convert the image to PDF
            c = canvas.Canvas(pdf_path, pagesize=img.size)
            c.drawImage(
                os.path.join(input_folder, file), 0, 0, img.size[0], img.size[1]
            )
            c.save()

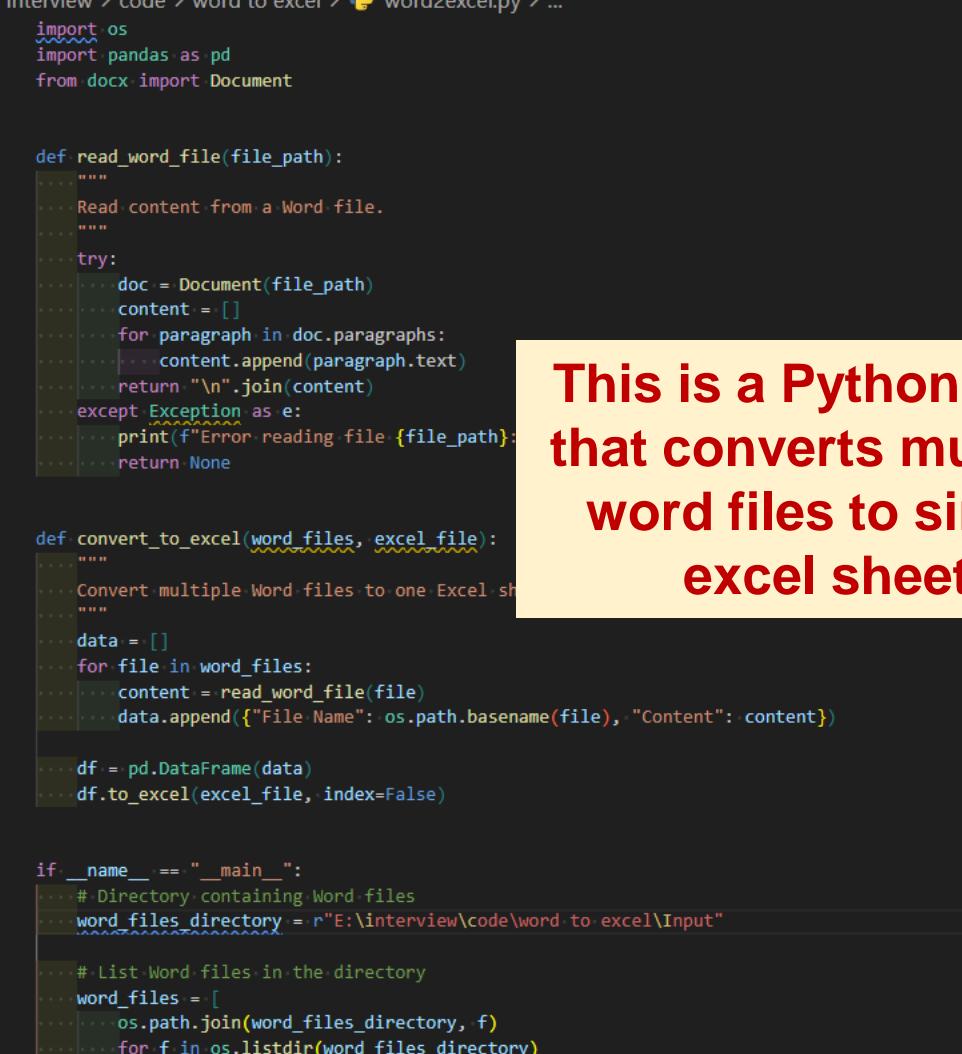
            print(f"Converted {file} to PDF")

# Provide the input folder containing JPEG images and the output folder to save PDFs
input_folder = r"E:\interview\code\JPEG To PDF\Input"
output_folder = r"E:\interview\code\JPEG To PDF\Output"

# Call the function to convert JPEG images to PDFs
convert_jpeg_to_pdf(input_folder, output_folder)
```

From this to this

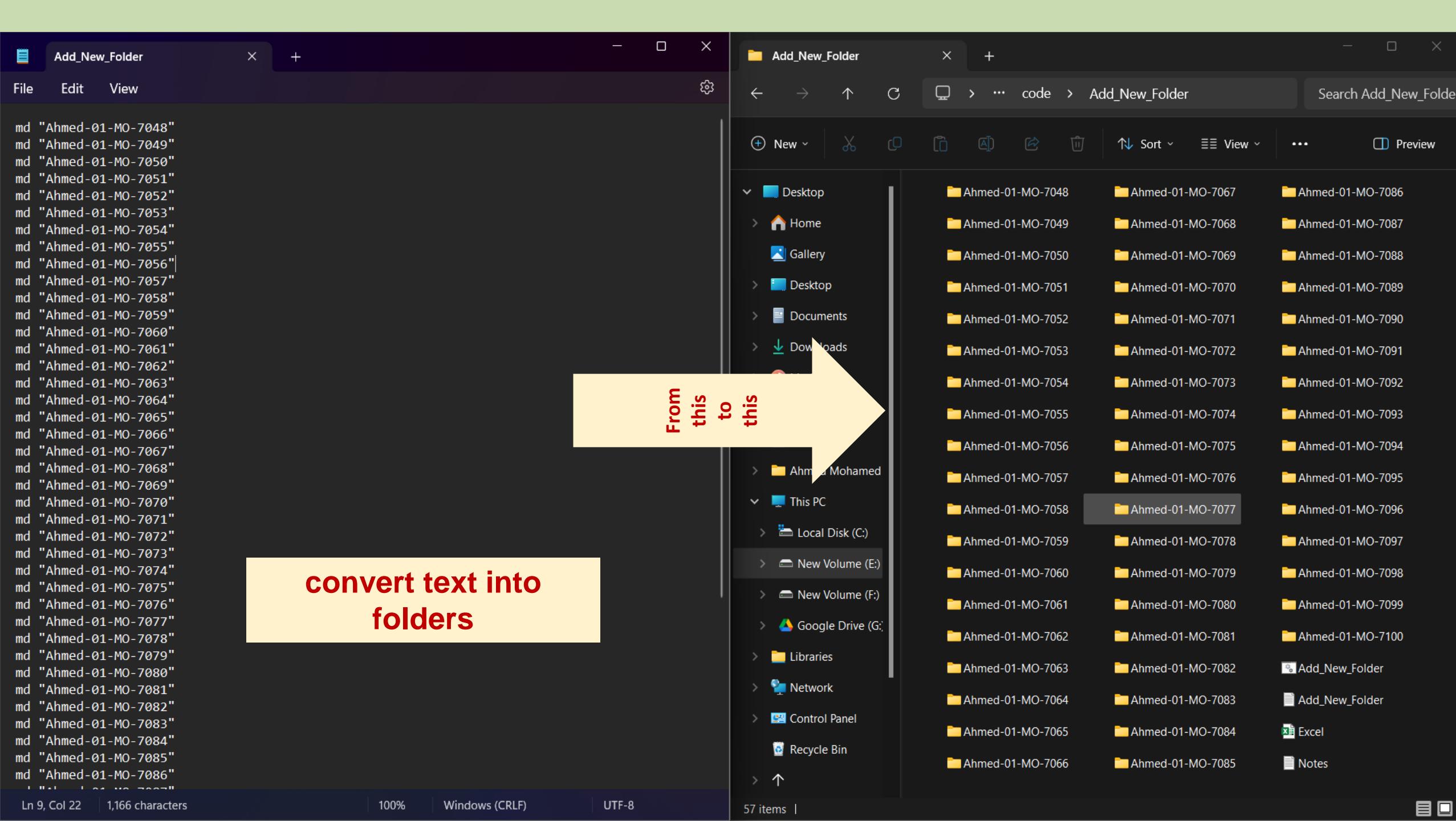
Name	Date modified	Type
AHMED-01-MO-1000	٢٠٢٤/١٢/١١ ١١:٥٣	PDF Document
AHMED-01-MO-1001	٢٠٢٤/١٢/١١ ١١:٥٣	PDF Document
AHMED-01-MO-1002	٢٠٢٤/١٢/١١ ١١:٥٣	PDF Document
AHMED-01-MO-1003	٢٠٢٤/١٢/١١ ١١:٥٣	PDF Document
AHMED-01-MO-1004	٢٠٢٤/١٢/١١ ١١:٥٣	PDF Document
AHMED-01-MO-1005	٢٠٢٤/١٢/١١ ١١:٥٣	PDF Document



E: > interview > code > word to excel > `word2excel.py` > ...

```
1 import os
2 import pandas as pd
3 from docx import Document
4
5
6 def read_word_file(file_path):
7     """
8         Read content from a Word file.
9     """
10    try:
11        doc = Document(file_path)
12        content = []
13        for paragraph in doc.paragraphs:
14            content.append(paragraph.text)
15        return "\n".join(content)
16    except Exception as e:
17        print(f"Error reading file {file_path}")
18        return None
19
20
21 def convert_to_excel(word_files, excel_file):
22     """
23         Convert multiple Word files to one Excel sheet.
24     """
25     data = []
26     for file in word_files:
27         content = read_word_file(file)
28         data.append({"File Name": os.path.basename(file), "Content": content})
29
30     df = pd.DataFrame(data)
31     df.to_excel(excel_file, index=False)
32
33
34 if __name__ == "__main__":
35     # Directory containing Word files
36     word_files_directory = r"E:\interview\code\word to excel\Input"
37
38     # List Word files in the directory
39     word_files = [
40         os.path.join(word_files_directory, f)
41         for f in os.listdir(word_files_directory)
42         if f.endswith(".docx")
43     ]
44
45     # Excel file to save the combined data
46     excel_file = (
47         r"C:\Users\Ahmed Mohamed\Desktop\code\word to excel\New Microsoft Excel Worksheet.xlsx"
48     )
```

This is a Python code
that converts multiple
word files to single
excel sheet





**Thank
You**