PYTHON FOR AUTOMATION

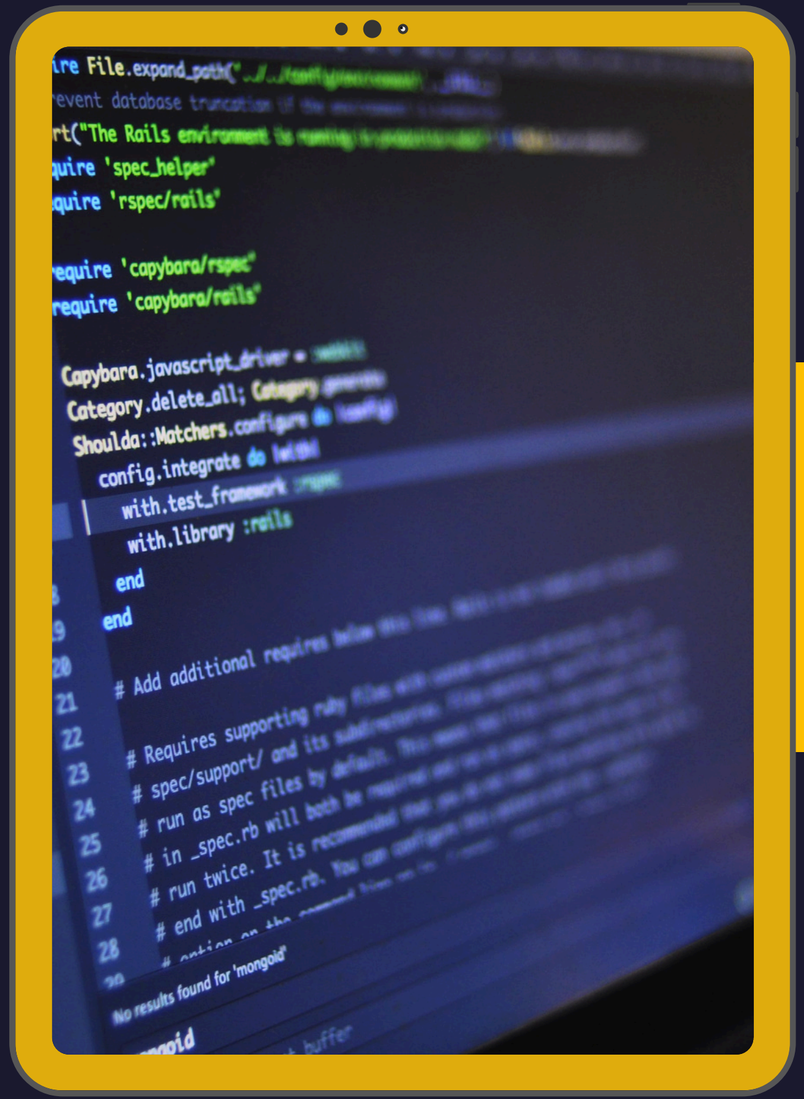
General Tasks Automation

PDSC Workshop Day 4

Sandesh Pyakurel

Overview

- Basics of Files I/O
- File paths
- OS module
- Rename images in a folder serially
- img2pdf module
- Convert Images to pdf
- qrcode module
- Customized QR code generator
- PIL
- Extension Conversions(png to jpg)
- Solution to assignment 2
- Assignment 3



Basics of Files I/O

What is File I/O?

- File I/O stands for File Input/Output.
- It involves operations like reading data from files and writing data to files.

Common file operations:

- Reading
- Writing
- Appending

Basics of Files I/O

Opening and Closing a file

```
f = open('path/to/the/file', 'mode')

# Reading or writing to the file

f.close()
```

Basics of Files I/O

Preferred way for file I/O

```
with open('path/to/the/file', 'mode') as f:
    # Reading or writing to the file
    f.read()
```

Modes

Read mode: “r”
Write mode: “w”
Append mode: “a”

Binary read mode: “rb”
Binary wirte mode: “wb”

File paths

- A file path is a string of characters which specifies the unique location of a file in a computer’s file system.

Absolute Path

- Absolute file paths specify the location of a file from the root directory in the file system.

Relative Path

- Relative file path specify the location of a file that is relative to the current directory.

File paths

Absolute path

```
/home/user/documents/file.txt
```

```
C:\Users\User\Documents\file.txt
```

Relative path

```
./reports/report.txt
```

```
.\invoices\invoice.pdf
```

- The . represents the current directory.
- The .. represents the parent directory.

OS module

- Python has a built-in OS module with methods for interacting with the operatiing system, like creating files and directories, management of files and directories and many more.

OS module

```python
import os

os.listdir('./report')
# ['text.txt', 'report.txt']

os.path.join('./reports', 'text.txt')
# './reports/text.txt'

os.rename('./report/text.txt', './report/report.txt')
```

```python
file_path = './report/text.txt'

os.path.splitext(file_path)
# ('./report/text', '.txt')
```

Renaming images in a folder serially

Why?

- Preparing Images for Machine Learning Models.
- Processing Images for E-commerce.

Renaming images in a folder serially

Create a virtual environment
and activate it.

Input: Images with random names in a folder

Output: Images with names 001_image.jpg, 002_image.jpg, and so on in the same folder.

Solution

Convert Images to PDF

Why?

- To compile related images to a same file.
- Sharing pdf is easier than sharing individual images.

img2pdf

It is an open source Python package to convert images to pdf format.

pip install img2pdf

img2pdf.convert(list_of_image_paths, layout)

Convert Images to PDF

Create a virtual environment and activate it.
Install img2pdf.

Input: A folder with images in it.

Output: Generate PDF file with the images in the folder.

Solution

qrcode

- A QR code (Quick Response code) is a type of matrix barcode (or two-dimensional barcode)
- It consists of black squares arranged on a white background, which can encode various types of information.

Features:

- Larger capacity
- Error correction
- Versatility
- Position Marker

qrcode

It is an open source Python package to generate qrcode.

```
pip install qrcode
```

```
qr = qrcode.QRCode( version, error_correction, box_size, border_size)

qr.add_data(data)
qr.make(fit=True)

img = qr.make_image(fill_color, back_color)

img.save()
```

qrcode

Solution

qrcode

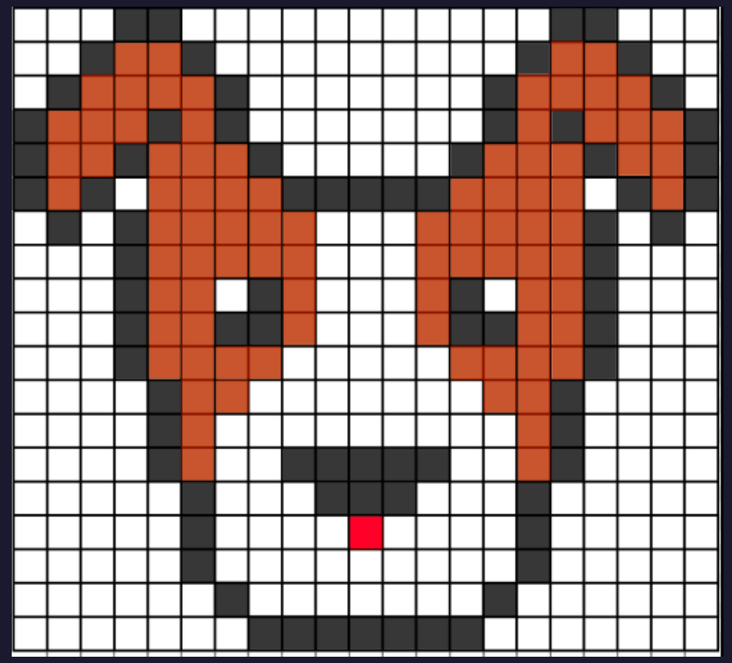
sengo library

PIL (Python Image Library)

- Python Imaging Library is an additional library for the Python Programming Language that adds support for opening, manipulating, and saving many different image file format.

- PNG

- JPG



PIL (Python Image Library)

- **PNG**

- Lossless compression
- RGBA
- High Quality

- **JPG**

- Lossy compression
- RGB
- Smaller file size

PIL (Python Image Library)

- **Why extension Conversion?**

- Png to jpg conversion for use in web applications.
- png to jpg conversion when there is size limits.

PIL (Python Image Library)

```
from PIL import Image

img = Image.open(image_path)

img.convert(“RGB”)

img.save()
```

Solution

Assignment (File compression with gzip)

- The gzip module in Python provides a simple interface for compressing and decompressing data using the GNU gzip file format.
- It is used for compressing single file.
- Compress level(1, 9) is used for speed vs depth trade-off.
- You can use tarfile module to compress entire folder.

THANK YOU