

[Get started](#)[Open in app](#)

towards
data science

[Follow](#)

594K Followers



Generating QR Codes With Python In Less Than 10 Lines

Constructing a simple Python Project For Encrypting Your Information With QR Codes



Bharath K 2 days ago · 8 min read★

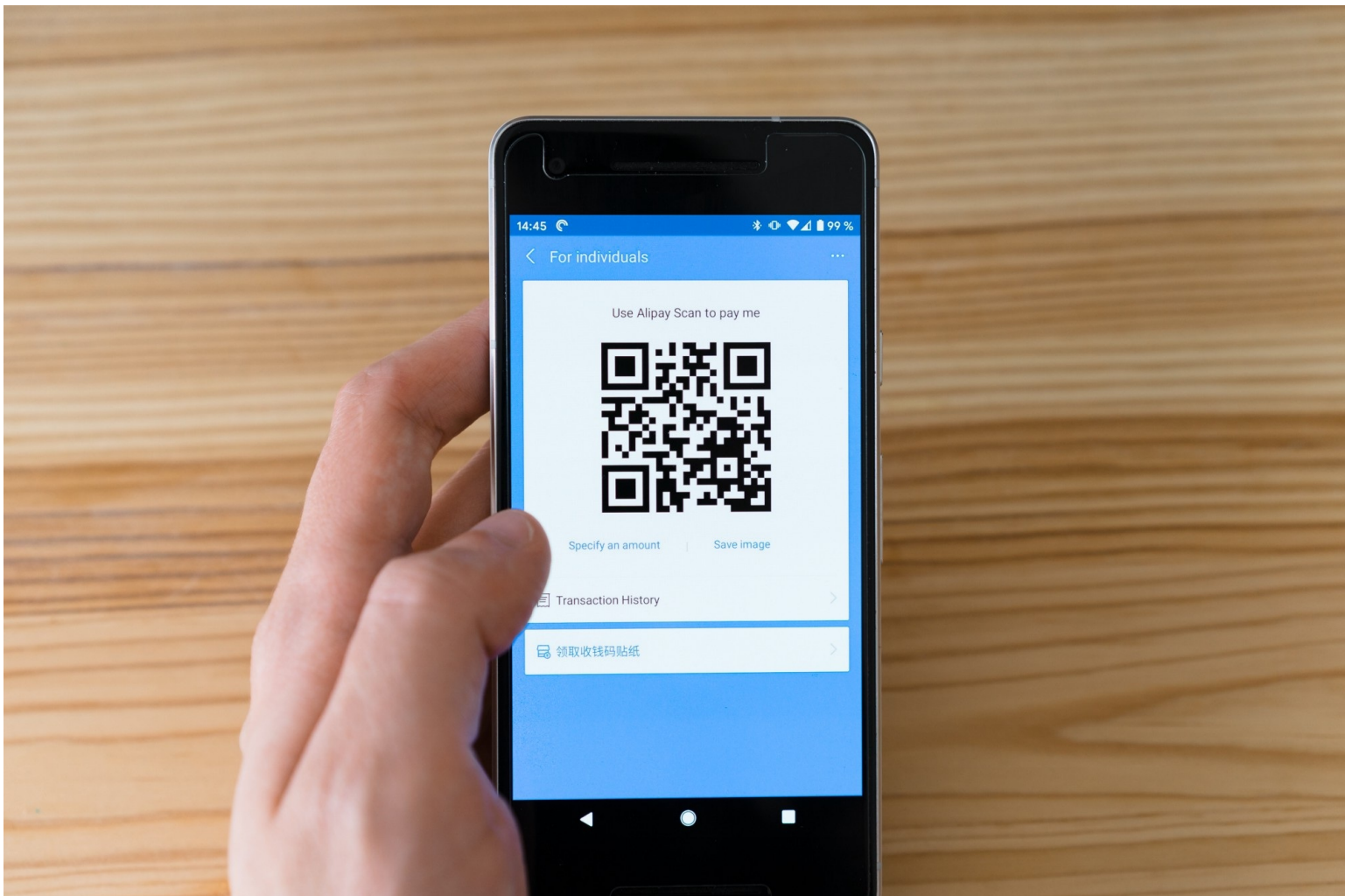


Photo by [Markus Winkler](#) on [Unsplash](#)

In the modern world, our objective is to always have a secure and convenient way of accessing things. Nobody wants to read and click on elongated URL links or lengthy word sequences. Also, in the world of the recent pandemic, it is usually considered best to avoid touches and achieve transactions without much physical contact.

This objective is achieved with the help of bar codes and QR codes. Bar codes suffer from some spacing limitations, which are handled by the introduction of QR codes. QR Codes are typically two-dimensional pictographic codes that offer the users a large storage capacity and fast readability in the form of black modules arranged in a square pattern on a white background.

QR codes are a fantastic resource for tracking information about numerous products, exchanging data, directing customers to a landing page or website, downloading apps, paying bills (at restaurants or other places), shopping, e-commerce, and so much more!

In this article, we will learn how we can utilize Python programming to create QR codes for any specific purpose. We will generate some bar codes for specific purposes and also find out how we can decode these generated codes with the help of some decoding steps. Finally, we will look at some of the additional stuff that you can accomplish with the encoding and decoding of QR codes.

. . .

Installing All Necessary Libraries:

For getting started with this project, there are a few library requirements in Python that we will ensure that we install successfully on our system to create, generate, and decode QR codes. The first library that we will install for encoding our QR codes is the QR code library.

The command to get started with the installation of the qrcode library module is quite simple and can be installed with the following pip command. You can add the following line of command either in the virtual environment of your choice or directly in the command prompt (or windows shell) if you have the path of the environment variables set accordingly.

```
pip install qrcode
```

Ensure that you also have the Pillow library installed in the particular Python environment. This library is one of the best options for operating with images and dealing with most functions related to visual aesthetics. The alternative command shown in the below command block will install both the essential libraries together with the additional pillow library for generating the images of the QR codes that will be stored.

```
pip install qrcode[pil]
```

The final library that we will require for the successful completion of this

project is the computer vision library, Open-CV. The following library is one of the more essential elements for most tasks related to image processing and computer vision. The purpose of this module for this project is to utilize one of its significant functions to decode the hidden information in the generated QR codes. The library is installable with the following command shown below.

```
pip install opencv-python
```

Computer vision is one of the most intriguing aspects of Data Science and Artificial Intelligence. The subject is on a constant boom and will continue to rise in popularity in the upcoming years. Hence, I would recommend viewers who are interested in this field check out my beginner's mastery guide to Open-CV for getting started with all the essential concepts of computer vision from the below link.

OpenCV: Complete Beginners Guide To Master the Basics Of Computer Vision With Code!

A tutorial with codes to master all the important concepts of computer vision and how to implement them using OpenCV

towardsdatascience.com



. . .

Encoding Your Secret QR Code:

In this section of the article, we will utilize the qrcode library to encode our information and generate a quick response code that we can access with some decoding. For generating the QR code, please follow the code snippet provided below, and you will have the image of the particular QR code with the relevant data stored in it.

```
import qrcode

# Create The variable to store the information
# link = "https://bharath-k1297.medium.com/membership"
data = "Hello! Welcome To World Of Programming!"

# Encode The Link
img = qrcode.make(data)
print(type(img))

# Save the QR Code
img.save("test1.jpg")
```

The first step to encode your desired code is to import the qrcode library that we previously installed. Once we import the library, we can either create a variable that stores the link to a particular website or create a variable that stores some useful information embedded in the form of a QR code. Encode the link into the form of an image and save the QR code that is now generated in a file name that you desire. When you print the

type or format of the image that is generated, you should notice the following class displayed —

```
<class 'qrcode.image.pil.PilImage'>
```

Now that we have finished learning how to encode the data in the form of QR code images, let us explore some of the ways in which the following data is decodable.

. . .

Decoding The Secret QR Code:



Image By Author

The above image representation is a representation of a QR code containing a link. If you are trying to access a QR code with a URL link, try to use a mobile app for decoding such QR codes or any other application to read the QR code. The above QR code, when read by an app, will direct you to the provided URL website. (The following link supports other authors and me if you apply for membership).

For decoding URL links such as the above image, you can make use of QR code scanners. However, if you are trying to access encrypted data, you can decode the information with the help of Open-CV. The QR Code Detector function provides the user with the option of decoding the particular QR code image and retrieving the stored data in the QR code. Check out the below code snippet for a brief idea on you can perform the following action.

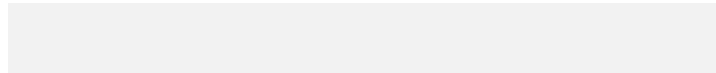
```
import cv2

# Create The Decoder
decoder = cv2.QRCodeDetector()
```

```
# Load Your Data
file_name = "test1.jpg"
image = cv2.imread(file_name)

# Decode and Print the required information
link, data_points, straight_qrcode = decoder.detectAndDecode(image)
print(link)
```

Here, we are importing the cv2 library and creating a variable that will act as the decoder for cracking the particular QR code image. We will then proceed to load the data and read it with the help of the cv2 library. We can then access the data and print the information accordingly. The below screenshot shows my output. Note that I am running the decoder code in the decode.py file in the tensors virtual environment.



. . .

Additional Building Blocks:

In this section, we will understand a couple of improvements that you can make to this Python project. Firstly, if you are developing your project for a specific purpose like a start-up or any other service or simply for fun, you can add a mini logo of the product, brand, or yourself as an integral part of the QR code to avoid confusion and have a clear representation of the item.

The next improvement that we can make to our project is to control the QR code that is generated. The following code block shown below demonstrates how such a process can be carried out accordingly. The version attribute determines the matrix size of the QR code. The error correction used for the QR Code scan is analyzed with the error connection attribute. The box size is the pixel size of each box, and the border controls the thickness. I would recommend experimenting with the below code block.

```
import qrcode

qr = qrcode.QRCode(
    version=1,
    error_correction=qrcode.constants.ERROR_CORRECT_L,
    box_size=10,
    border=4,
)
```

. . .

Conclusion:



Photo by [Toa Heftiba](#) on [Unsplash](#)

QR codes are extremely significant in the modern world today, where most of the transactions and useful data are conveyed with minimal physical contact in the most technical mechanism. Whether you want to exchange information, conduct transactions, or simply specify the essential URL links to a particular website, these QR codes are a high utility resource.

In this article, we understood how easy it is to generate QR codes with the help of Python programming in less than ten lines of code. With the installation of the required libraries, you can generate your own QR codes and decode them accordingly. You can embed useful URL links or important information in these QR codes and convey it to others in a simplistic, highly structured format.

If you want to get notified about my articles as soon as they go up, check out the following [link](#) to subscribe for email recommendations. If you wish to support other authors and me, then subscribe to the below link.

Join Medium with my referral link - Bharath K

As a Medium member, a portion of your membership fee goes to writers you read, and you get full access to every

bharath-k1297.medium.com



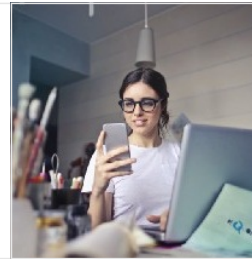
The technology of QR codes is quite revolutionary, and it will continue to exist until a better discovery replaces it. Hence, it is a great time to start exploring this topic. If you have any queries related to the various points stated in this article, then feel free to let me know in the comments below. I will try to get back to you with a response as soon as possible.

Check out some of my other articles that you might enjoy reading!

How To Write Code Effectively In Python

Analyzing the best practices that you should follow while writing code in Python

towardsdatascience.com



5 Best Python Projects With Codes That You Can Complete Within An Hour!

Constructing five cool Python projects with complete codes that developers can build within an hour to add to your...

towardsdatascience.com



7 Best UI Graphics Tools For Python Developers With Starter Codes

The seven best UI graphics tools available in Python for developing cool user interface technologies

towardsdatascience.com



Thank you all for sticking on till the end. I hope all of you enjoyed reading the article. Wish you all a wonderful day!

Artificial Intelligence

Python

Programming

Technology

Productivity