This code implements a digital image watermarking system using the Discrete Cosine Transform (DCT) for embedding and extracting hidden messages in images. The watermarking is performed in the frequency domain by manipulating the DCT coefficients of the image.

* Importing Libraries:
  + cv2: OpenCV library for computer vision tasks.
  + numpy: Library for numerical operations.
  + types: Unused in this code.
  + base64: Encoding and decoding base64 strings.
  + qrcode: Generating QR codes.
  + zxing: Barcode reader library.
  + matplotlib.pyplot: Plotting library.
  + io and PIL.Image: Input/output and image processing libraries.
  + scipy.fftpack: Fast Fourier Transform functions.
  + cryptography: Cryptography library for RSA encryption.
* File Paths:
  + Constants PRIVATE\_KEY\_FILE\_PATH and PUBLIC\_KEY\_FILE\_PATH store file paths for the private and public keys.
* Key Generation and Loading:
  + Functions to generate an RSA key pair (generate\_rsa\_key\_pair and generate\_and\_save\_rsa\_key\_pair) and load the keys from disk (load\_rsa\_key\_pair\_from\_disk).
* RSA Encryption and Decryption:
  + encrypt\_message: Encrypts a message using RSA encryption.
  + decrypt\_message: Decrypts an RSA-encrypted message.
* Base64 Encoding and Decoding:
  + base64\_to\_image: Converts a base64-encoded image back to a file.
  + image\_to\_base64: Converts an image to base64 encoding.
* QR Code Generation and Decoding:
  + generate\_qr\_code: Generates a QR code from a message.
  + generate\_qr\_code\_with\_image: Embeds an image and a message into a QR code.
  + decode\_qr\_with\_zxing: Decodes a QR code using ZXing.
  + decode\_qr\_code\_with\_image: Extracts the image and message from a QR code.
* DCT and Watermarking Functions:
  + hideDataDCT: Embeds a binary message into the DCT coefficients of an image.
  + showDataDCT: Extracts a binary message from the DCT coefficients of an image.
  + encode\_text\_dct: Prompts the user to input image and message for embedding.
  + decode\_text\_dct: Prompts the user to input a watermarked image for message extraction.
* Quality Metrics:
  + psnr: Peak Signal-to-Noise Ratio calculation.
  + ncc: Normalized Cross-Correlation calculation.
* Main Embedding and Extraction Function:
  + LSB: Main function that prompts the user to choose between embedding or extracting data.
* Key Existence Check and Main Execution:
  + Checks if the key files exist and generates/loads keys accordingly.
  + Calls the LSB function to either embed or extract data.