

Cryptography



What is our GOAL for this CLASS?

In this class, we learned about cryptography, techniques used in Cryptography

What did we ACHIEVE in the class TODAY?

- We Understood about Cryptography and Cryptanalysis.

Which CONCEPTS/ CODING BLOCKS did we cover today?

- We used the libraries, smtplib,ssl
- We used key events to capture

Understanding concepts:

What is **Cryptography**?

Cryptography is the study and application of techniques that hide the real meaning of information by transforming it into nonhuman readable formats and vice versa.

Encryption

Encryption is the process of converting information into a form that is not readable by humans. The encrypted information is known as a ciphertext.

Decryption

Decryption is done using a secret key, which is only known to the recipients of the information.

A key is required to decrypt the hidden messages. By doing so, even if a hacker obtains the information, it will no longer make sense to them.

Symmetric key algorithms (Private key cryptography):

A symmetric encryption is one that uses the same key to encrypt and decrypt data. One example of this is the Caesar Cipher.

Asymmetric key algorithms (Public key cryptography) Each party has a private key (kept secret) and a public key (known to all).

How did we DO the activities?

1. Create one function main() which will ask for user input which function want to perform encryption ,decryption
 - Print ("Choose one option)
 - Initialize variable choice which will ask for input encryption decryption
 - If choice == 1 , perform encryption
 - If choice ==2 perform decryption
 - Else print("wrong choice")

```
def main():  
    print()  
    print("Choose one option")  
    choice = int(input("1. Encryption\n2. Decryption\nChoose(1,2): "))  
    if choice == 1:  
        encryption()  
    elif choice == 2:  
        decryption()  
    else:  
        print("Wrong Choice")
```

2. Create function name **encryption()**

- **Print ("encryption")**
- Variable **msg** which will save input from user Variable **key** will save required shift range from **(1-94)** from user
- Use for loop and check the length of the **"msg"**
- Create variable **temp** , which will store ASCII value of character. **ord()** function returns the Unicode code from a given character. This function accepts a string of unit length as an argument and returns the Unicode equivalence of the passed argument.
- If **temp** is **greater than 126**, 126 are total no of ASCII characters then check the user input number which is store in **temp subtract 127** and add **32 shift**
- Create variable **encrypted_text** which will store **temp** value after converting into character again The **chr()** method returns a string representing a character whose Unicode code point is an integer
- **print ("encrypted text")**
- Call the **main()** function

```
def encryption():
    print("Encryption")
    msg = input("Enter your message: ")
    key = int(input("Enter key(1-94): ")) # based on 26 letters of alphabet

    encrypted_text = ""

    for i in range(len(msg)):
        temp = (ord(msg[i]) + key)
        if temp > 126:
            temp = temp - 127 + 32
        encrypted_text += chr(temp)

    print("Encrypted: " + encrypted_text)

main()
```

3. Create function name **decryption()**

- **Print ("decryption")**
- Variable **encryp_msg** which will save encrypted value which user want to decrypt
- Variable **decryp_key** will save required shift range from **(1-94)** from user
- Use for loop and check the length of the **"encryp_msg"**
- Create variable **temp** , which will store ASCII value of character. **ord()** function

returns the Unicode code from a given character and then subtract the same shift value i.e decrypt_key

- If **temp** is less than 32, check the user **input number** which is store in **temp** add 127 and subtract 32 shift
- Create variable **decrypted_text** which will store **temp** value after converting into character again
- The **chr()** method returns a string representing a character whose Unicode code point is an integer.
- **print ("decrypt text")**

```
def decryption():  
  
    print("Decryption")  
  
    print("Message can only be Lower or Uppercase alphabet")  
    encrp_msg = input("Enter encrypted Text: ")  
    decrp_key = int(input("Enter key (1-94): "))  
  
    decrypted_text = ""  
  
    for i in range(len(encrp_msg)):  
        temp = (ord(encrp_msg[i]) - decrp_key)  
        if(temp < 32):  
            temp = temp + 127 - 32  
  
        decrypted_text += chr(temp)  
  
    print("Decrypted Text: " + decrypted_text)
```

What's NEXT?

In the next class we will learn about steganography

Expand Your Knowledge

To know more about [click here](#)