**MORSE CODE TRANSLATOR**

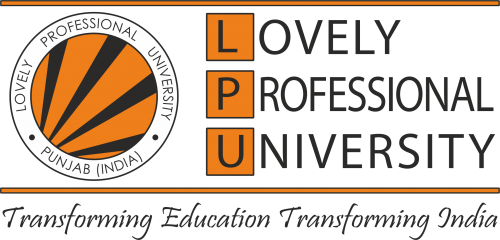
**SUBMITTED By: - SUBMITTED To: -**

**Kushal Anjana Sagar Pande Sir**

**Anushil Sharma**

**REG. NO: 11809796 Section: - K18pv**

**REG. NO: 11809795**

****

**LOVELY PROFESSIONAL UNIVERISTY**

**PHAGWARA,PUNJAB**

**MORSE CODE TRANSLATOR**

**Introduction: -**

From the time of world war and from generations people use to send there messages in the certain formats to their friends and alliance which can’t be decoded or predicted by any third party.

This method help them to send there important messages during the time of war and the message which we send can be encoded with the help of Morse code.

**Aim :-**

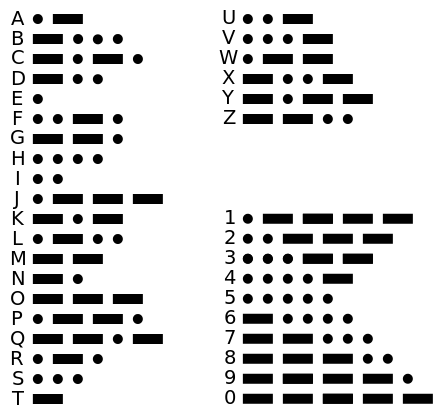
Our major aim was to built a user-friendly interface of Morse code translator where a person can encode as well as decode their respective messages, can get the information about Morse code.

They can also give suggestion and feedback regarding to the application.

**About Morse code: -**

**Morse code** is a character encoding scheme used in telecommunication that encodes text characters as standardized sequences of two different signal durations called *dots* and *dashes* or *dits* and *dahs*. Morse code is named for Samuel F.B.Morse, an inventor of the telegraph.

**Basic Morse code**



The basic Morse code consist of English alphabets from A-Z

and basic numbers from 0-9.

for each character there is a specific sequence of dash and dots.

**Motivation: -**

Our Motivate to make a Morse code application to decode all types of characters.

English alphabets

English numerals

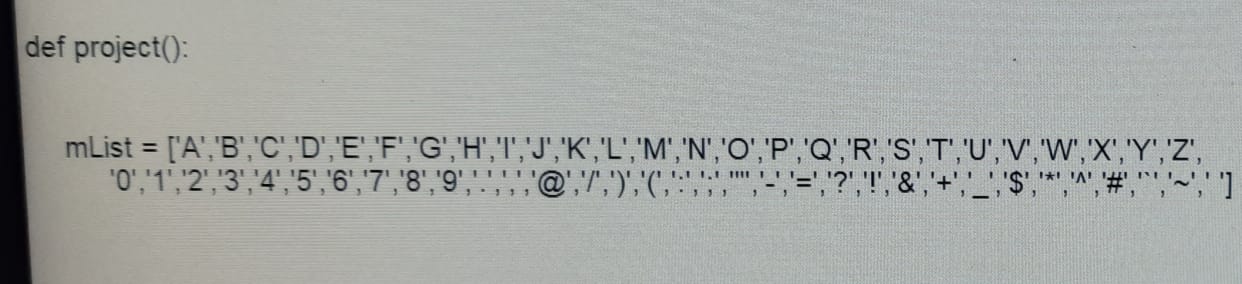
Special symbols

We get our motive from the Morse code used by the army during the time of world wars.

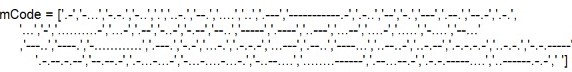
**Objective: -**

**1.**Major objective is to describe and encode as well decode those special symbols also which are not used in basic Morse code. Our list of special characters: -

Sequence of Decryption list: -



Sequence of Encryption list: -



**2.** Our second major objective was to use Dynamic Memory allocation in Dictionary which was use to store character for Decryption and Encryption. To manage the memory allocation at the run time for character going to be defined in dictionary.

**3.** Third objective was to create a suggestion box for any suggestions or feedbacks for the application and Morse code.

**4.** Fourth objective was to give information about Morse code when it was created and give you all the information to learn which help you further in learning Morse code.

**Contribution: -**

Team members two: -

Kushal Anjana- Graphic user interface building, code logic, Plotting images

Anushil Sharma- Presentation, Report, Suggestion Box

**Contents of the project: -**

1. **About Morse Code**
2. **Learn Morse Code**
3. **Encryption**
4. **Decryption**
5. **Suggestion Box**
6. **Exit**

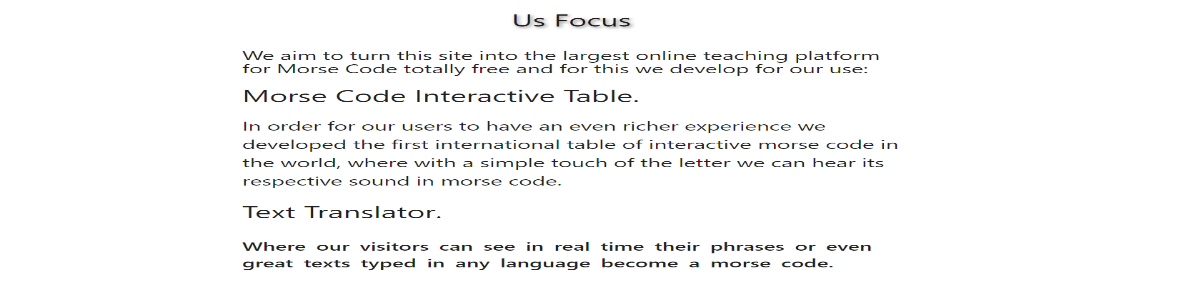
**About Morse code: -**

In this sub topic we gather the information about Morse code its uses and how to use by gathering the information from the official site of Morse code and getting the details about Morse code directly by linking its URL

URL = "https://www.morsecodeninja.com/index.html"

webbrowser. open(url)

Glimpse of website

****

**Learn Morse Code: -**

In this sub topic we gather the information about Morse code learning and its uses and how to use it.

It gives the information about: -

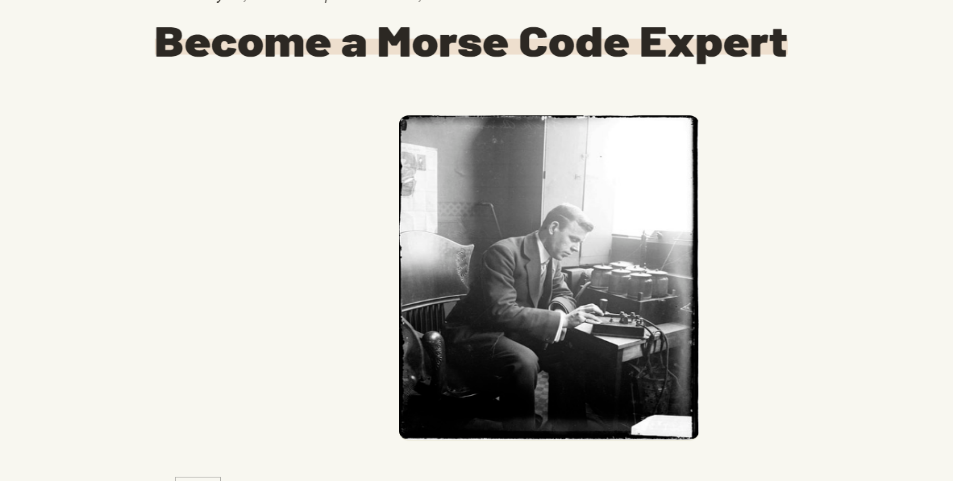
**1. How to get familiar with Morse code**

**2. Tips to make memorization of Morse code Easier**

**3. Use of nifty chart for help in learning Morse code**

**4. By listening to actual Morse code sound**

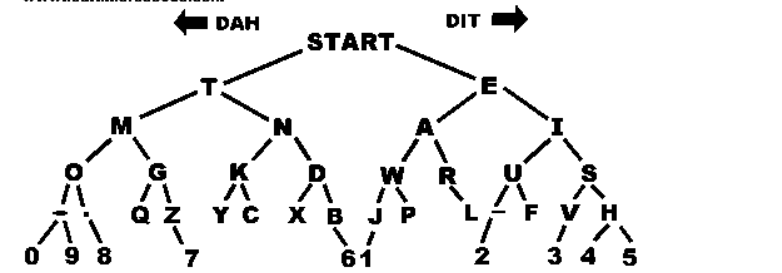
Glimpse of website



**Nifty Chart: -**

This is a code listening tool. Print it on your printer.  
Place your pencil where it says START and listen to Morse code.  
Move down and to the right every time you hear a DIT (a dot).  
Move down and to the left every time you hear a DAH (a dash).

**Nifty Chart diagram**

****

For getting the details about how to learn Morse code we gather this information by linking our application from url as follow: -

URL “https://www.morsecodeninja.com/index.html" webbrowser.open(url)

**Encryption: -**

morse\_code\_dict={}

for i in range(0,len(mList)):

morse\_code\_dict[mList[i]]=mCode[i]

msg = tkinter.Text(win,height=2,width=23)

def morse\_this():

text\_message=text.get().upper()

morse\_code\_list = []

for each\_char in text\_message:

for morse\_key,morse\_value in morse\_code\_dict.items():

if each\_char == morse\_key:

morse\_code\_list.append(morse\_value)

morse\_code\_string=' '.join(morse\_code\_list)

Explanation: -

Firstly, we made an empty dictionary name as morse\_code\_dict

In which we do dynamic memory allocation for the Morse code characters to allocate memory at the run time for them.

Then the both lists which are created for encryption and decryption are linked by a for loop and each value of list of Morse code is assigned to the character list which are going to encrypted.

Then we created msg variable for the output box for encryption of the text that we entered in text box which is created by variable name text\_message .

The nested for loops is created for each\_char entered in text\_message and morse\_key and morse\_value in morse\_code\_dict.If each\_char is equal to morse\_key so,Now by creating an empty list to store morse\_value with the help of append function then we print the encrypted code in the output box with the help of morse\_code\_string variable and then passing it as argument in msg variable for output.

msg.insert(tkinter.END,morse\_code\_string)

**Decryption: -**

m = tkinter.Text(win,height=2,width=23)

def decrypt\_this():

st=' '

text\_message=text.get().upper()

for i in text\_message.split():

for key,value in morse\_code\_dict.items():

if value==i:

st+=key

Explanation: -

we created m variable for the output box for decryption of the text that we entered in text box which is created by variable name text\_message .

Firstly, create a function as decrypt\_this and a empty string as st for entering the values

Now by nested for loop for I in text\_message and key,values in morse\_code\_dict. If value is equal to i the st =st+key.

Then we print the decrypted code in the output box with the help of st variable and then passing it as argument in m variable for output.

m.insert(tkinter.END,st)

**Suggestion box: -**

In suggestion box we have use following syntax

Suggestion box <heading>

Full name🡪<enter name>

Email ID🡪 <enter email>

Gender🡪 <select>

Country🡪 <select>

Suggestion🡪<enter message>

In suggestion box we created six labels for: -

Heading, Full name, Email, Gender, Country, Suggestion

In suggestion box we created three entry boxes: -

Full name, Email, Suggestion box





We take the entire data of list1 in the droplist in string format

**Modules used: -**

* **webbrowser**
* **win sound**
* **sqlite3**

**Webbrowser: -**

Used in modules like: -

* About Morse code
* Learn Morse code

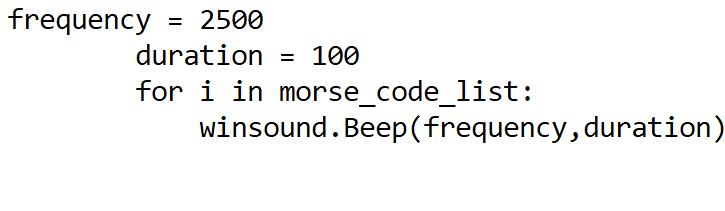
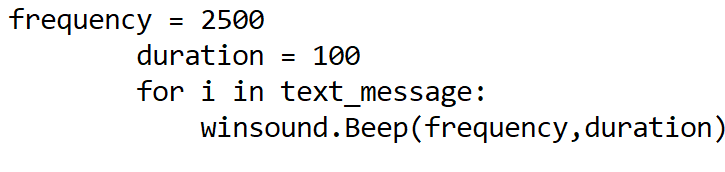
To use hyperlinks to gather the accurate and specific information about Morse code and about learning it by connecting my project with the internet.

**Win sound: -**

Used in module of Translation: -

For Encryption as well as Decryption of the input text or message.

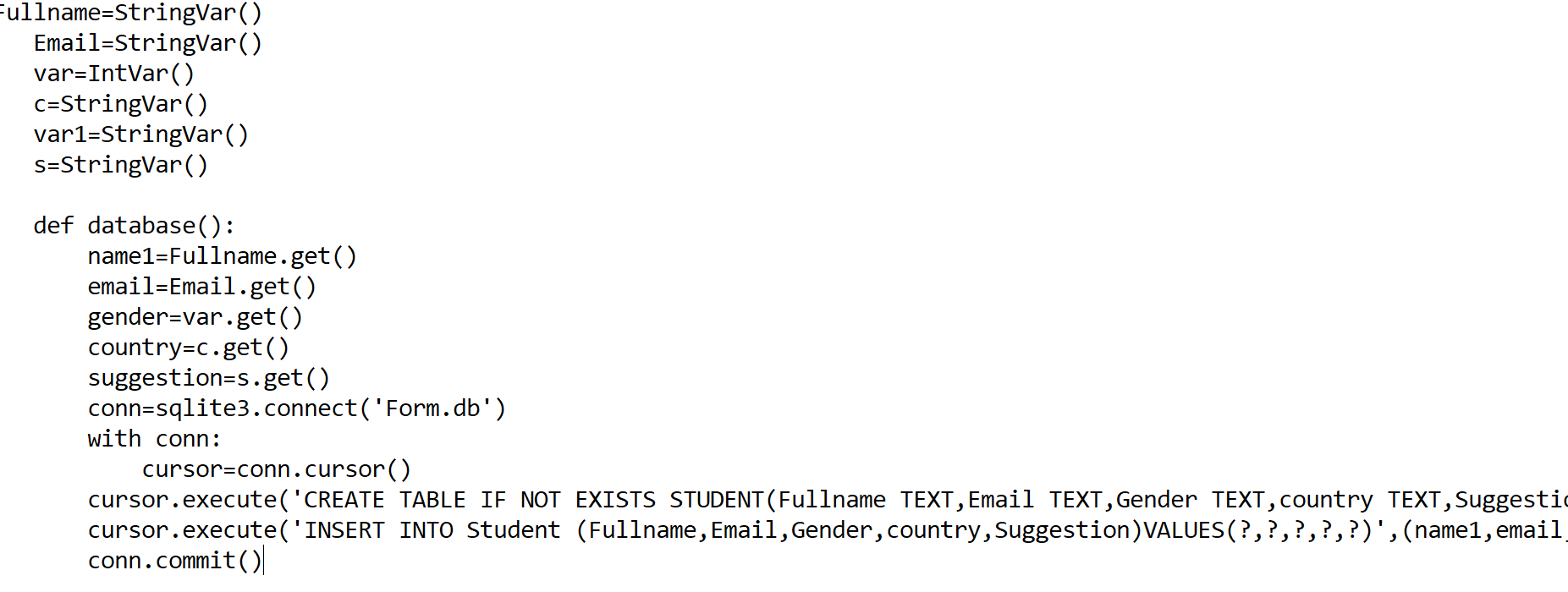
Encryption code: - Decryption code:-

**Sqlite3 Or SQLITE: -**

We have use Sql lite for creating the data base of our suggestion page for a proper record of suggestions and data given by people.

So, basically creates its own server for connecting the database to the Morse code Interface.



Firstly we defined the type of each and every variable

Then created a function as database() and get all the variables defined outside the function

Which were further used in creating the tables.

**Libraries used: -**

Only one library is used to built a perfect Graphic User Interface which is Known as Tkinter.

Tkinter: - **Tkinter** is the standard GUI library for **Python**. **Python** when combined with **Tkinter** provides a fast and easy way to create GUI applications. **Tkinter** provides a powerful object-oriented interface to the Tk GUI toolkit.

**Use of Tkinter in the project: -**

* Firstly, we import the Tkinter library in our application code.
* Then imported all the features of it by using (import Tkinter \*)
* Tk() class is used to create a root window where the Graphic user interface will run.

**Tags used: -**

1. Image tags
2. Button tags
3. Labels tags
4. Entry tags

**Image Syntax: -**

photo = PhotoImage(file = "name")

w = Label(root, image=photo)

w.place(“coordinates”, anchor=”position”)

**Button Syntax: -**

pic=PhotoImage(fill name)

<variable name> = pic.subsample(“coordinates")

btn = tkinter.Button(root,image = <variable name>)

btn.configure(background="color", “size”)

btn.place(“coordinates”, anchor=”position”)

**Label Syntax: -**

label = Label(root, text="FullName",size,font=("bold"))

label.place(coordinates)

**Entry Syntax: -**

entry = Entry(root,Variable type)

entry.place(coordinates)

**We have created total three GUI windows: -**

* Interface
* Suggestion box
* Encryption and Decryption

**References: -**

For suggestion box we took reference form Youtube for connecting the database

[**https://www.youtube.com/watch?v=YP2FKRZNfY0**](https://www.youtube.com/watch?v=YP2FKRZNfY0)

For the basic help in Morse code translator

[**https://www.youtube.com/watch?v=-unqE1a0-HU**](https://www.youtube.com/watch?v=-unqE1a0-HU)

