DAILY ASSESSMENT

Date:	**27-5-2020**	**Name:**	**Kavyashree m**
Course:	**Digital signal processing**	**USN:**	**4al15ec036**
Topic:	**Fourier transform, FFT,FFT-fast fourier transform MatLab,FIR and IIR filters ,study and analysis FIR and IIR using FDA tool in MatLab,Introduction to WT , CWT & DWT , implementing of signal using WT in MatLab ,short-time fourier transform and the spectrogram.Welch's method and windowing ,ECG signal Analysis using MatLab.**	**Semester & Section:**	**8th A**
Github Repository:	**Kavya**		

FORENOON SESSION DETAILS

Image of session

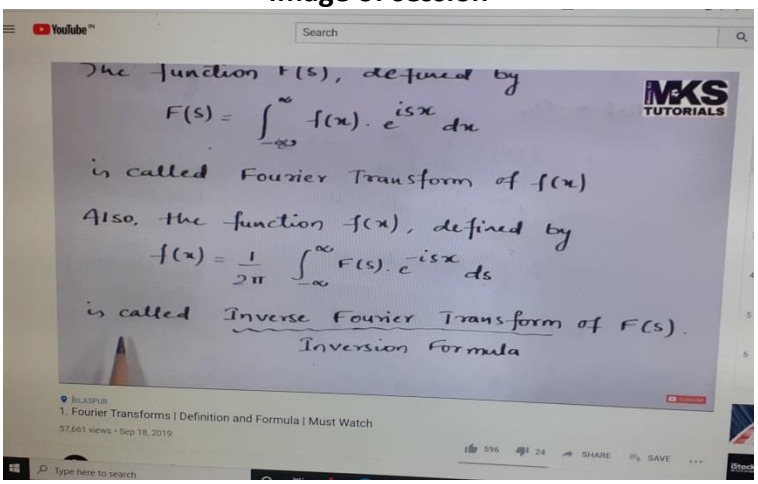


Fig 1: fourier transform

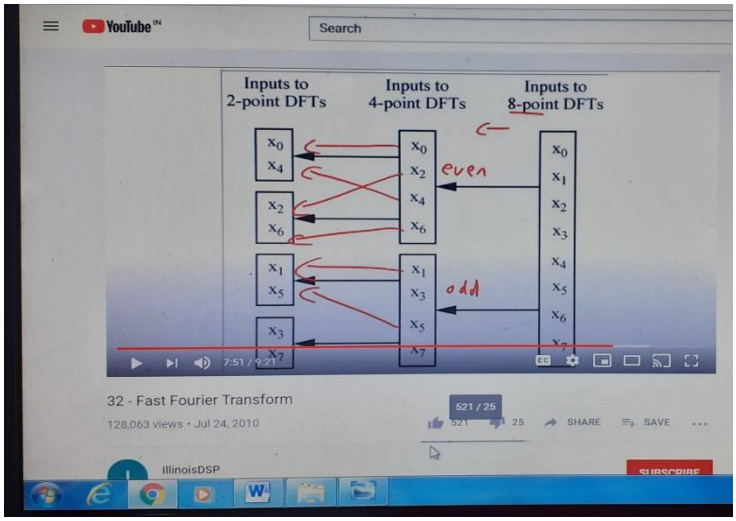


Fig 2 : FFT

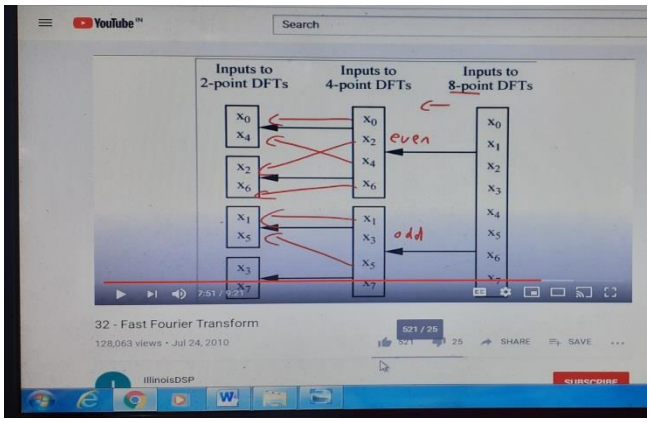


Fig 3: fast fourier transform MatLab

Fig 4 : FIR and IIR filters

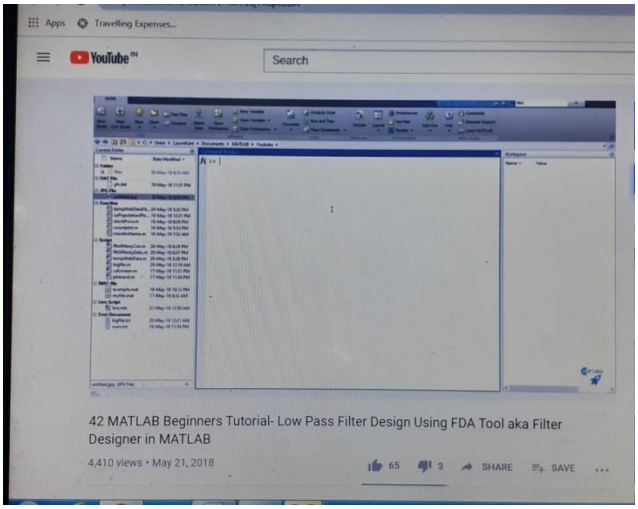


Fig 5: study and analysis FIR and IIR using FDA tool in MatLab

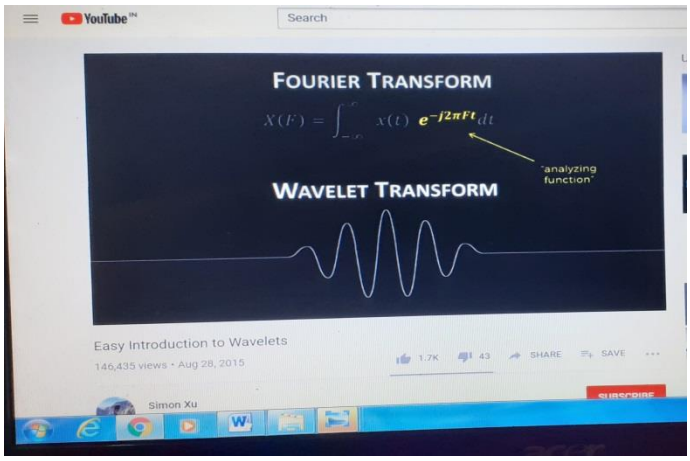


Fig 6 : Introduction to WT

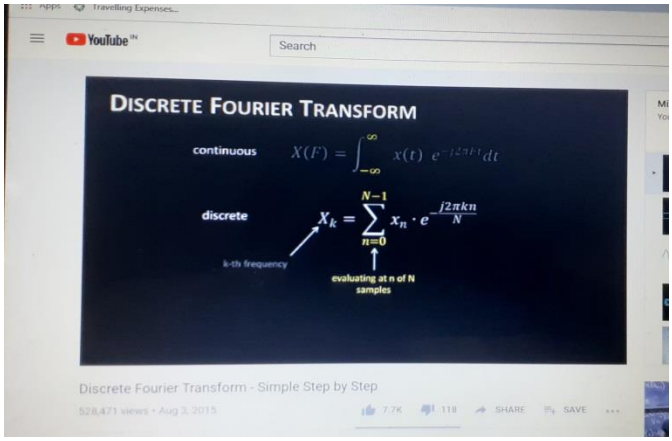


Fig 7: CWT & DWT

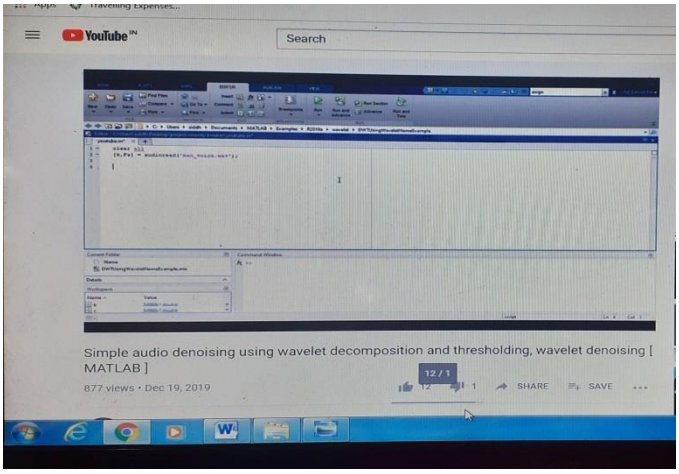


Fig 8: Implementing of signal using WT in MatLab

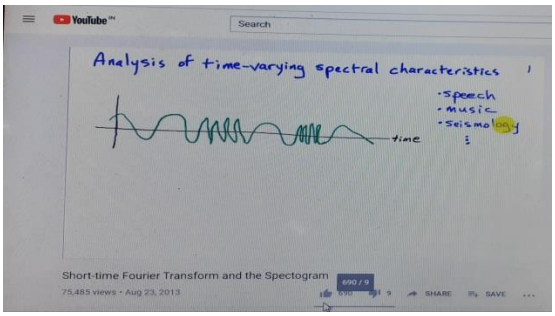


Fig 9 : short-time fourier transform and the spectrogram

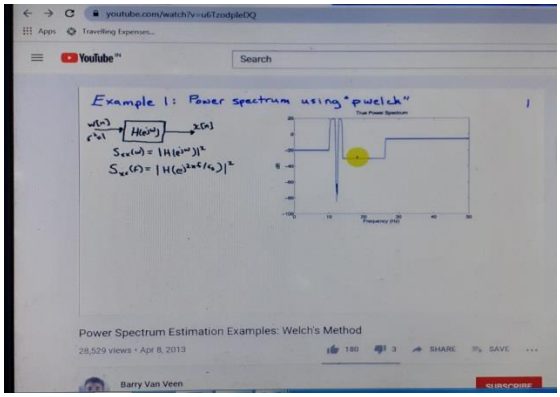


Fig 10: Welch's method and windowing

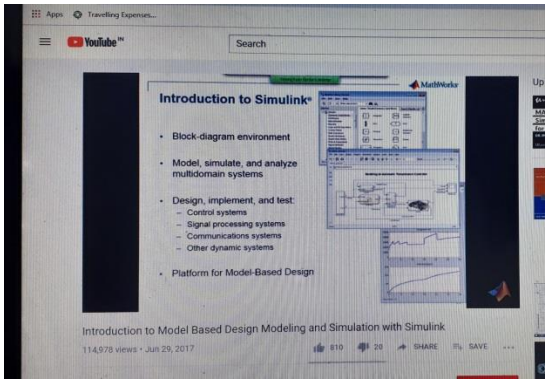


Fig 11 : ECG signal Analysis using MatLab.

Fourier transform

A Fourier transform (FT) is a mathematical transform which decomposes a function into its constituent frequencies, such as the expression of a musical chord in terms of the volumes and frequencies of its constituent notes. The term Fourier transform refers to both the frequency domain representation and the mathematical operation that associates the frequency domain representation to a function of time.

FFT-fast fourier transform MatLab

MATLAB is a numerical computing environment developed by MathWorks. MATLAB allows matrix manip , ulations, p g lotting of functions and data, and implementation of algorithms Getting help You can get help by typing the commands help or lookfor at the >> prompt, e.g. >> help fft.

Data Representations in MATLAB Variables:

- Variables are defined as the assignment operator "=" . The syntax of variable assignment is vari bl a e name = a value (or an expressi) on For example, >> x = 5 x = 5 >> y = [3*7, pi/3]; % pi is in MATLAB Vectors/Matrices MATLAB can create and manip late arra s of 1 (ectors) 2 π
- Vectors/Matrices: MATLAB can create and manipulate arrays of 1 (vectors), 2 (matrices), or more dimensions row vectors: a = [1, 2, 3, 4] is a 1X4 matrix column vectors: b = [5; 6; 7; 8; 9] is a 5X1 matrix, e.g. >> A = [1 2 3; 7 8 9; 4 5 6] A = 123 789 4 5 6

FIR and IIR filters

Two classes of digital filters are Finite Impulse Response (FIR) and Infinite Impulse Response (IIR).The term 'Impulse Response' refers to the appearance of the filter in the time domain. Filters typically have broad frequency responses, which correspond to short duration pulses in the time domain as shown in Figure below .

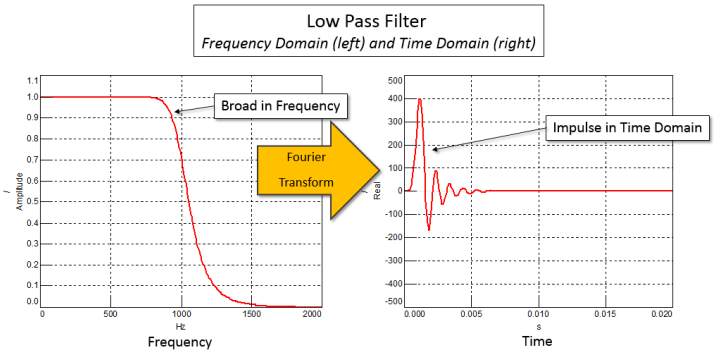


Figure a: Frequency (left) and time (right) domain representations of filter shape.

The equations for both an IIR and FIR filter are shown in Equation 1. The input to the filter is time series x(n), and the output of the filter is a time series y(n). The first sample

in the time history is at n=0.

FIR Filter Equation: $y(n) = \sum_{k=0}^{N} a(k)x(n-k)$

IIR Filter Equation: $y(n) = \sum_{k=0}^{N} a(k)x(n-k) + \sum_{j=0}^{P} b(j)y(n-j)$

Output used recursively

Equation 1: Finite Impulse Response (FIR) filter equations versus Infinite Impulse Response (IIR) filter.

The mathematical difference between the IIR and FIR implementation is that the IIR filter uses some of the filter output as input. This makes the IIR filter a 'recursive' function.

Each equation has three series of numbers: an input time history, a filter, and an output time history

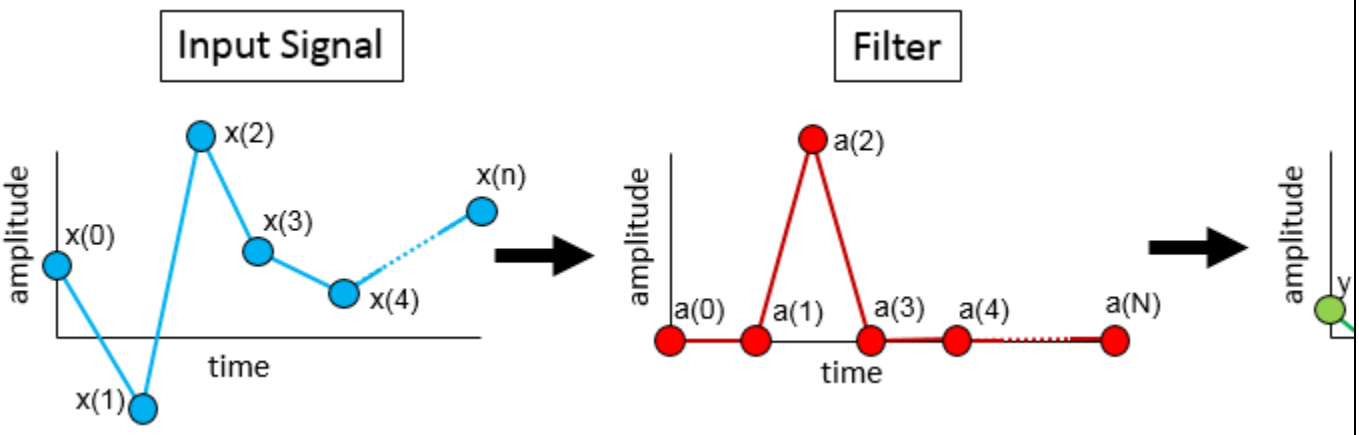


Figure b: Time series for input time history, "x(n)" , FIR filter - Series "a(k)", and output time history, "y(n)".

The series are:

- Series x(n) - The input time history is x(0), x(1), x(2), to x(n). Lower case n is the total number of data points in the input time history.
- Series a(k) - The filter is represented by the series labelled "a" for a FIR filter, and "a" and "b" for an IIR filter. Uppercase letters N and P are the number of terms in the filters respectively, which is also referred to as the filter "order" or "number of

taps". For example, the FIR filter is performed from k=0 to k=N.

- Series y(n) - Output time history is y(0), y(1), y(2), …

Study and Analysis FIR and IIR using FDA tool in MatLab

❖ MATLAB FDATOOL

MATLAB FDATOOL is a filter design analysis tool used mainly to design the filter on MATLAB platform and then transforms it into corresponding VHDL code. In the following section are the steps how to configure the FIR filter on MATLAB**.**

The implementation of fir filter requires the basic three

building blocks viz.

1) Multiplication

2) Adder

3) Signal delay

Introduction to WT

Wt is a C++ library for developing web applications. Admitted, C++ doesn't come to mind as the first choice for a programming language when one talks about web development. Web development is usually associated with scripting languages, and is usually implemented at the level of generating responses for incoming requests. Since both requests and responses are text encodings, web programming is ultimately a text processing task, and thus conveniently expressed in a scripting language.

CWT & DWT

- The CWT and the discrete wavelet transforms differ in how they discretize the scale parameter. The CWT typically uses exponential scales with a base smaller than 2, for example $2^{1/12}$. The discrete wavelet transform always uses exponential scales with the base equal to 2. The scales in the discrete wavelet transform are powers of 2. Keep in mind that the physical interpretation of scales for both the

CWT and discrete wavelet transforms requires the inclusion of the signal's sampling interval if it is not equal to one.

- The decimated and nondecimated discrete wavelet transforms differ in how they discretize the translation parameter. The decimated discrete wavelet transform (DWT), always translates by an integer multiple of the scale, $2^j m$. The nondecimated discrete wavelet transform translates by integer shifts.

Welch's method and windowing

Welch's method estimating power spectra is carried out by dividing the time signal into successive blocks, forming the periodogram for each block, and averaging.

Denote the m th windowed, zero-padded frame from the signal x by

$$x_m(n) \stackrel{\Delta}{=} w(n)x(n+mR), \quad n = 0, 1, \ldots, M-1, \; m = 0, 1, \ldots, K-1,$$

where R is defined as the window hop size, and let K denote the number of available frames. Then the periodogram of the m th block is given by

$$P_{x_m,M}(\omega_k) = \frac{1}{M} \left|\mathrm{FFT}_{N,k}(x_m)\right|^2 \stackrel{\Delta}{=} \frac{1}{M} \left| \sum_{n=0}^{N-1} x_m(n) e^{-j2\pi nk/N} \right|^2$$

as before, and the Welch estimate of the power spectral density is given by

$$\hat{S}_x^W(\omega_k) \stackrel{\Delta}{=} \frac{1}{K} \sum_{m=0}^{K-1} P_{x_m,M}(\omega_k).$$

In other words, it's just an average of periodograms across time. When $w(n)$ is the rectangular window, the periodograms are formed from non-overlapping successive blocks of data.

ECG signal Analysis using MatLab.

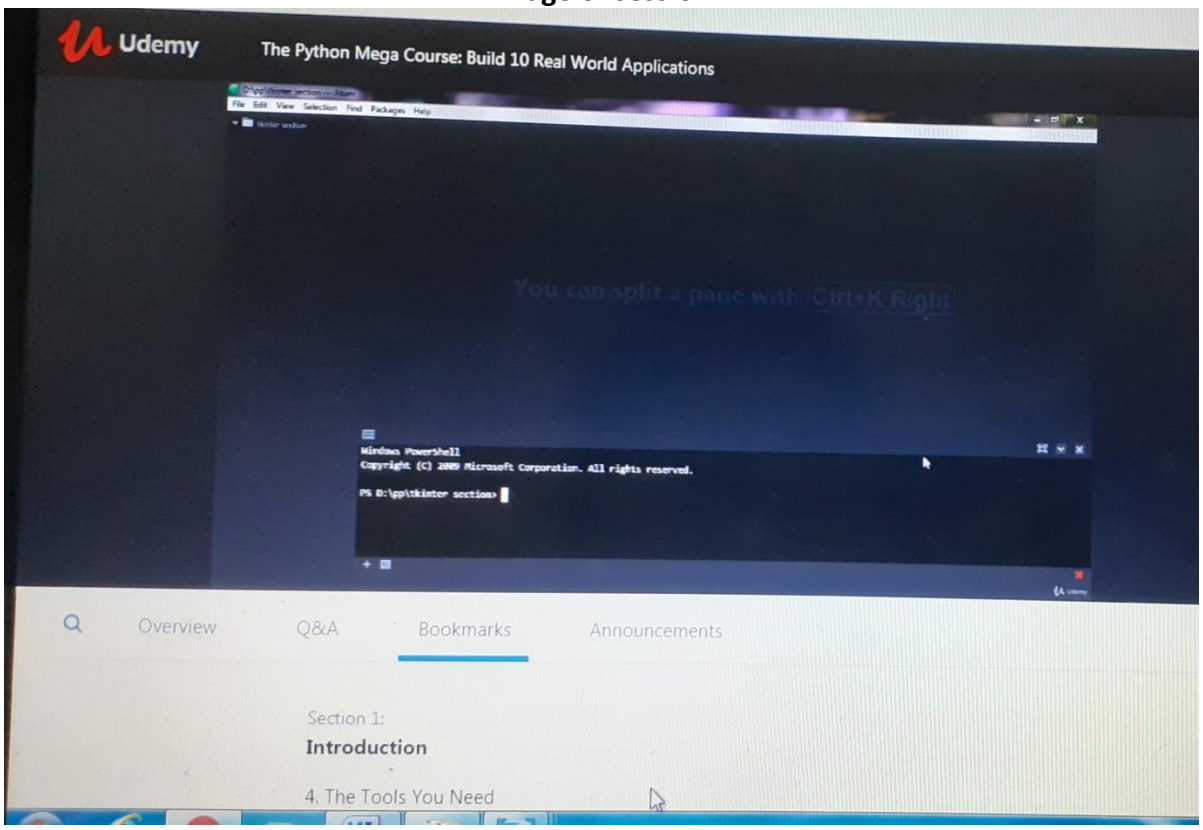
Electrocardiogram is a diagnostic tool that measures and records the electrical activity of

the heart in exquisite detail. Interpretation of these details allows diagnosis of a wide range of heart conditions. These conditions can vary from minor to life threatening An ECG is generated by a nerve impulse stimulus to a heart. The current is diffused around the surface of the body surface. The current at the body surface will build on the voltage drop, which is a couple of μV to mV with an impulse variation. Usually, this is very small amplitude of impulse, which requires a couple of thousand times of amplification. A typical ECG tracing of a normal heartbeat consists of a P wave, a QRS complex and a T wave. A small U wave is normally visible in 50 to 75% of ECGs. The baseline voltage of the electrocardiogram is known as the isoelectric line. Typically the isoelectric line is measured as the portion of the tracing following the T wave and preceding the next P wave. The electrical activity of the heart can be recorded at the surface of the body using an electrocardiogram. Therefore the electro-cardio-gram (EKG) is simply a voltmeter that uses up to 12 different leads (electrodes) placed on designated areas of the body. The electrical activity of the heart is generally sensed by monitoring electrodes placed on the skin surface. The electrical signal is very small (normally 0.0001 to 0.003 volt). These signals are within the frequency range of 0.05 to 100 Hertz (Hz.) or cycles per second.

Date:	**27-5-2020**	**Name:**	**Kavyashree m**
Course:	**Python programming**	**USN:**	**4al15ec036**
Topic:	**Graphical user interfaces with TKinter ,Interfaces with databases**	**Semester & Section:**	**8th A**
Github Repository:	**kavya**		

AFTERNOON SESSION DETAILS

Image of session



Graphical user interfaces with TKinter

The tkinter package is a thin object-oriented layer on top of Tcl/Tk. To use tkinter, you don't need to write Tcl code, but you will need to consult the Tk documentation, and occasionally the Tcl documentation. tkinter is a set of wrappers that implement the Tk

widgets as Python classes. In addition, the internal module _tkinter provides a threadsafe mechanism which allows Python and Tcl to interact.

Setting up a GUI with Widgets

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.

Creating a GUI application using Tkinter is an easy task. All you need to do is perform the following steps −

- Import the *Tkinter* module.
- Create the GUI application main window.
- Add one or more of the above-mentioned widgets to the GUI application.
- Enter the main event loop to take action against each event triggered by the user.

Example

```
#!/usr/bin/python

import Tkinter
top = Tkinter.Tk()
# Code to add widgets will go here...
top.mainloop()
```

This would create a following window −

Tkinter Widgets

Tkinter provides various controls, such as buttons, labels and text boxes used in a GUI application. These controls are commonly called widgets.

Connecting GUI widgets with call back Functions

A callback is a function memory reference that is called by another function and that takes the first function as a parameter. Put simply, a callback is a function that you provide to another function so that it can calling it.

If a callback does not take any argument, it can be handled with a simple function, such as the one shown in the preceding code. However, if a callback needs to take arguments, we can use the lambda function, as shown in the following code snippet:

Copy

```
def my_callback (argument)
  #do something with argument
```

Create a multi-widget GUI

Modern computer applications are user-friendly. User interaction is not restricted to console-based I/O. They have a more ergonomic graphical user interface (GUI) thanks to high speed processors and powerful graphics hardware. These applications can

receive inputs through mouse clicks and can enable the user to choose from alternatives with the help of radio buttons, dropdown lists, and other GUI elements (or widgets).

The following example creates a window with a button, label and entry field.

Example: Create Widgets

```
from tkinter import *
window=Tk()
btn=Button(window, text="This is Button widget", fg='blue')
btn.place(x=80, y=100)
lbl=Label(window, text="This is Label widget", fg='red', font=("Helvetica", 16))
lbl.place(x=60, y=50)
txtfld=Entry(window, text="This is Entry Widget", bd=5)
txtfld.place(x=80, y=150)
window.title('Hello Python')
window.geometry("300x200+10+10")
window.mainloop()
```

Interfaces with databases

Python with database

The Python standard for database interfaces is the Python DB-API. Most Python database interfaces adhere to this standard.You can choose the right database for your application. Python Database API supports a wide range of database servers such as −

- GadFly
- mSQL
- MySQL
- PostgreSQL
- Microsoft SQL Server 2000
- Informix

- Interbase
- Oracle
- Sybase

Introduction to PostgreSQL Psycopg2

MySQLdb is an interface for connecting to a MySQL database server from Python. It implements the Python Database API v2.0 and is built on top of the MySQL C API.

How do I Install MySQLdb?

Before proceeding, you make sure you have MySQLdb installed on your machine. Just type the following in your Python script and execute it −

```
#!/usr/bin/python
import MySQLdb
```

If it produces the following result, then it means MySQLdb module is not installed −

```
Traceback (most recent call last):
  File "test.py", line 3, in <module>
    import MySQLdb
ImportError: No module named MySQLdb
```

To install MySQLdb module, use the following command −

For Ubuntu, use the following command -

```
$ sudo apt-get install python-pip python-dev libmysqlclient-dev
```

For Fedora, use the following command -

```
$ sudo dnf install python python-devel mysql-devel redhat-rpm-config gcc
```

For Python command prompt, use the following command -

```
pip install MySQL-python
```

Quering data from a MySQL

The SQL SELECT command is used to fetch data from the MySQL database. You can use this command at mysql> prompt as well as in any script like PHP.

Syntax

Here is generic SQL syntax of SELECT command to fetch data from the MySQL table –

```
SELECT field1, field2,...fieldN
FROM table_name1, table_name2...
[WHERE Clause]
[OFFSET M ][LIMIT N]
```

- You can use one or more tables separated by comma to include various conditions using a WHERE clause, but the WHERE clause is an optional part of the SELECT command.
- You can fetch one or more fields in a single SELECT command.
- You can specify star (*) in place of fields. In this case, SELECT will return all the fields.
- You can specify any condition using the WHERE clause.
- You can specify an offset using OFFSET from where SELECT will start returning records. By default, the offset starts at zero.
- You can limit the number of returns using the LIMIT attribute.