Object Oriented Methodology

Week – *9*, Lecture – *2*

Basics of UI Design

SAURABH SRIVASTAVA

VISITING FACULTY

IIIT LUCKNOW

The Main Window

All Window Applications have a Main Window

This Window is the main Container where other UI elements reside

Usually, it is possible to create secondary Windows that are “attached” to the Main Window

- The prime example is a Dialog

The secondary Windows may be *modal* or *non-modal*

- A modal secondary Window does not let the user interact with the Main Window …
- … i.e., the user must close the secondary Window, before they can interact with the Main Window
- A non-modal Window appears like an independent Window …
- … i.e., the user can interact with it independently

The decision to pick a modal or non modal window depends on the use case

Widgets and Containers

Widgets are different types of UI elements that you can add in a Window

Some examples are

- Label – A widget to display text or images
- Button – A “clickable” widget which is often used to trigger events
- Textbox – A single line input widget to take textual input
- Textarea – A multiple-line textbox
- Scrollbars – Usually added to a Container widget, allows scrolling through the content of the container
- Others – Dropdown Lists, Checkboxes, Radio boxes etc.

Container widgets are used to group multiple widgets into one group

- It is usually done to apply a different *layout* strategy

Layout Strategy

A Layout strategy decides how widgets are shown within a Container or the Main Window

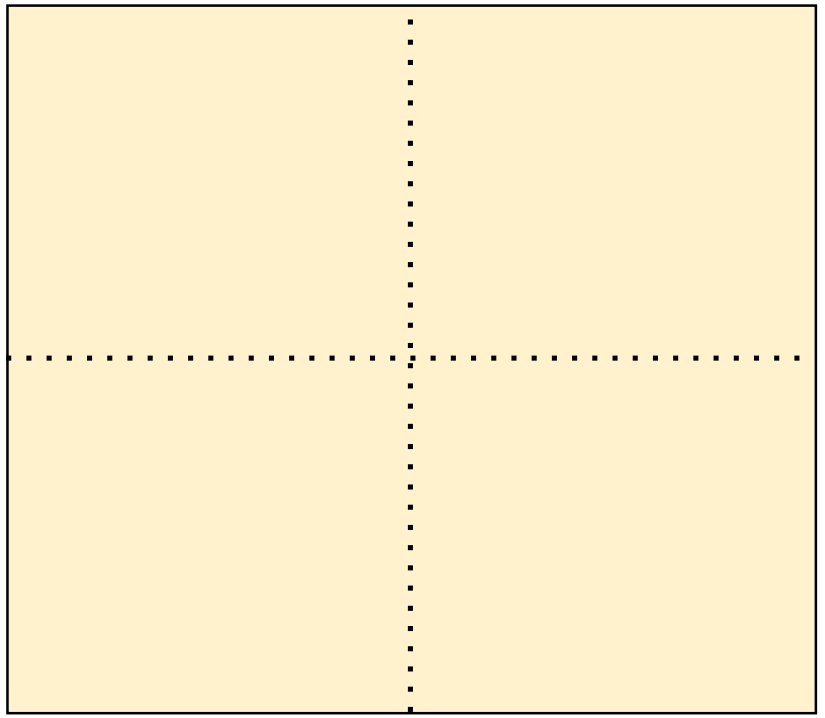
Some of the decisions which are made as part of this strategy include

- The position of different widgets in the container, relative to other widgets
- The minimum and maximum size that a widget can occupy on the screen
- Whether a widget is resized or not, if its container is resized …
- … also, what happens to the extra or reduced space within the container

Usually, most UI libraries provide some pre-defined layout strategies

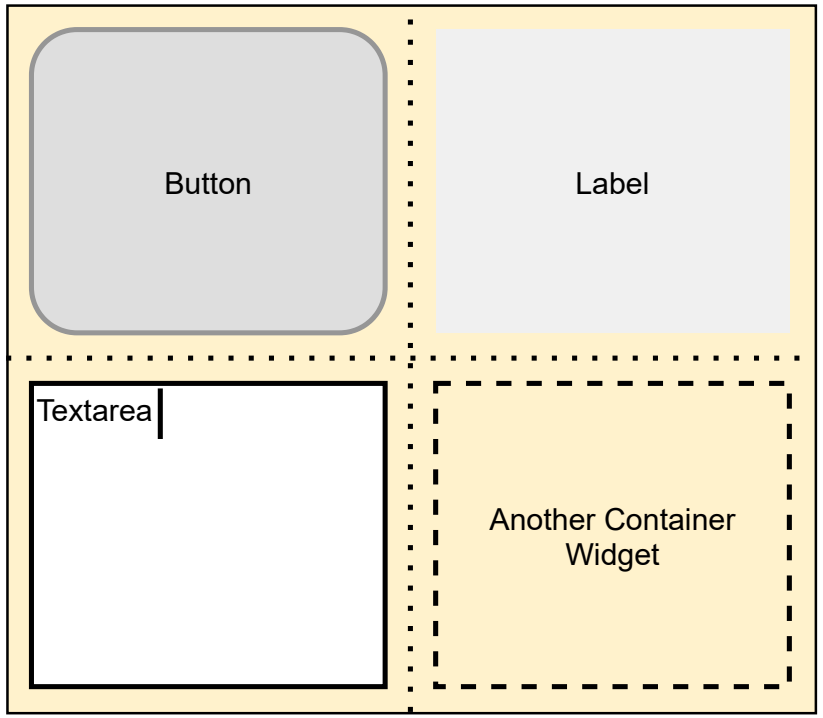
- The most common one is Grid Layout, which lays the widgets in the container inside invisible grids

Example – Grid Layout



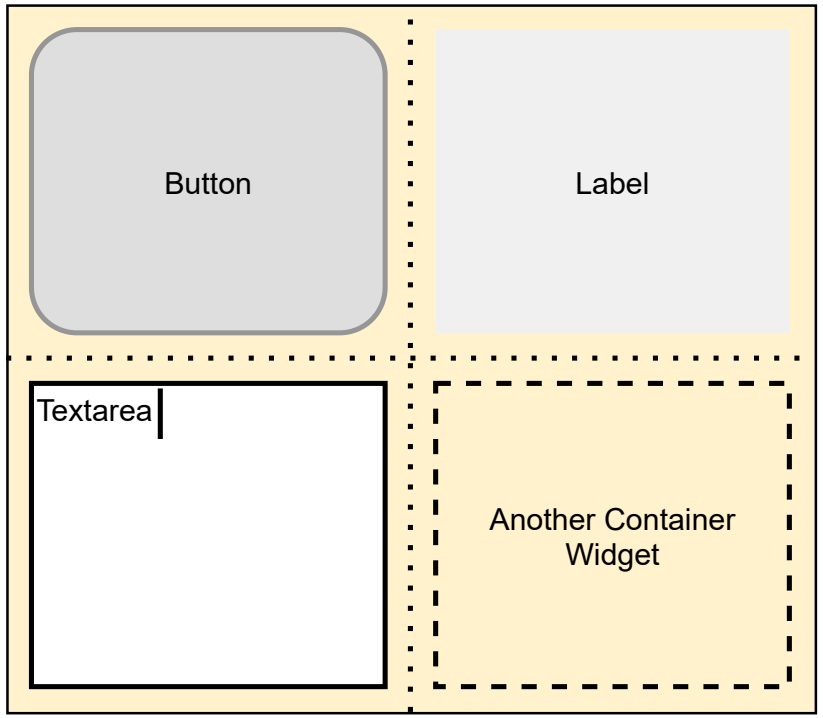
A 2x2 grid for laying out components

Example – Grid Layout



A 2x2 Grid Layout with components

Example – Grid Layout



A 2x2 Grid Layout with components

The spacing between the components is usually configurable (often defaulting to 0 units)

UI Events and Event Handlers

The common programming methodology for building UI applications is to respond to *UI Events*

A UI event could be as complex as the user dragging and dropping something on the Window …

- … or as simple as the movement of the mouse within a specific part of the Window

For the UI events that interests us, we are supposed to write *Event Handlers*

Usually, the underlying UI toolkit allows us to provide handlers for a wide range of events

- The most common event that is often handled is the *click* event
- For example, you would like certain processing to happen when the user clicks on a specific button
- Other examples of events include user typing a key on the keyboard or minimising the Main Window

The Menu Bar and Taskbar

Some UI toolkits provide two special container widgets added to the Main Window

The Menu Bar can contain a set of Menus (e.g. File, Edit, Help etc.)

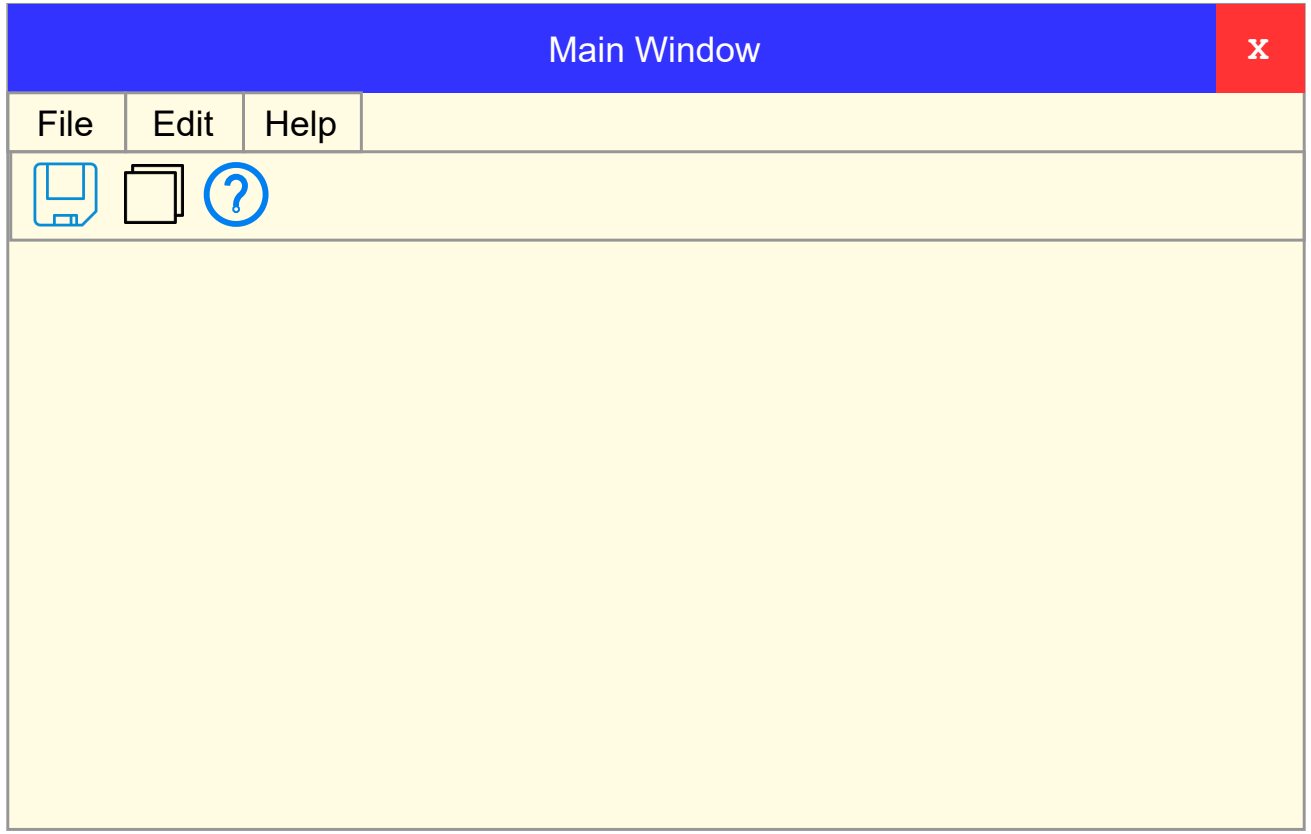
Each Menu, in turn can have Menu items or Menu item collections

Another widget that some UI toolkits provide is a Taskbar

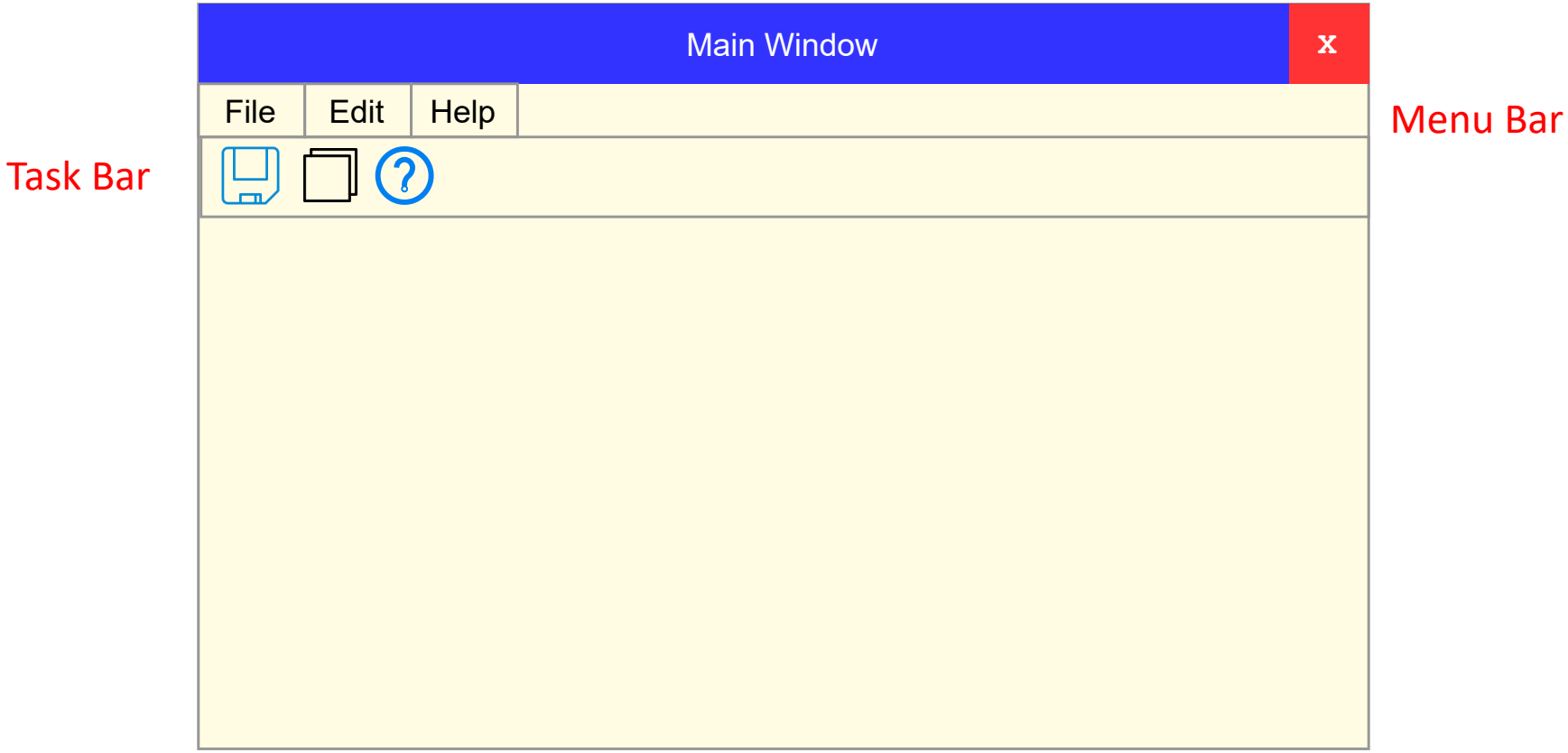
- It is usually placed immediately below the Menu Bar

It can be used to show icons which can be short-cuts for common tasks

Example – Menu Bar and Task Bar



Example – Menu Bar and Task Bar



Installing Qt

Qt is a cross-platform UI Toolkit for making Window Applications in C++

There are two versions – the Open Source version and the Commercial version

We will use the Open Source version to get an idea of UI design

You need to create an account for yourself on their website

- Go here for the same: https://login.qt.io/register

You can download the online installer for Qt from here:

- https://www.qt.io/download-open-source

Run the installer and select Custom Installation option

- Then, select the “Desktop gcc 64-bit” and “Qt Creator…” options for installation and go ahead

You may need to install `OpenGL` as well; on Ubuntu 21.04, the following command works:

- `sudo apt install libgl1-mesa-dev`

Homework !!

Install **Qt** on your machine

- You can also do so on your Mac or Windows installation