Java GUI Programming

Exploring AWT Components: Frame, Panel, Button, Choice, MenuBar, and the Legacy of Applets.

What is Java AWT?

🖥 **AWT:** Stands for Abstract Window Toolkit. It's Java's original, platform-dependent API for creating Graphical User Interfaces (GUIs).

📦 **Components:** AWT provides basic components like buttons, lists, and menus that rely on the native operating system's (OS) components.

🗇 **Containers:** It also provides containers like `Frame` and `Panel` to hold and organize these components within a window.

</> **Foundation:** It is the foundational library for Java GUI programming and serves as the base for the more modern Swing library.

The Legacy of Java Applets

An applet was a small Java program designed to be embedded in a web page and run inside the browser's Java Virtual Machine (JVM).

- They were popular in the early web for adding interactive content.
- **Deprecated:** Due to major security risks and lack of modern browser support, Applets were deprecated in Java 9.
- **Obsolete:** They are no longer used in modern web development, which relies on JavaScript, HTML5, and frameworks.

Deprecated or Obsolete Technology Icon

AWT Containers

The 'Boxes' That Hold Your GUI

Frame Class Program

What is a Frame?

A `Frame` is a top-level window with a title bar, border, and (optionally) a menu bar.

It's the primary container for most AWT-based desktop applications. Think of it as the main window of your program.

Example Code

```java
import java.awt.*;

public class FrameExample {
    public static void main(String[] args) {
        // Create a new Frame
        Frame f = new Frame("My First Frame");

        // Set its size (width, height)
        f.setSize(300, 200);

        // Make the frame visible
        f.setVisible(true);
    }
}
```

Panel Class Program

What is a Panel?

A `Panel` is a generic, simple container. It has no title bar or border.

Its main purpose is to group other components together (like buttons, labels) inside a `Frame` or another `Panel` to help organize the layout.

Example Code (Adding to Frame)

```java
import java.awt.*;

public class PanelExample {
    public static void main(String[] args) {
        Frame f = new Frame("Frame with Panel");
        Panel p = new Panel();

        // Set panel background to see it
        p.setBackground(Color.lightGray);

        // Add a button to the panel
        p.add(new Button("Click Me!"));

        f.add(p); // Add the panel to the frame
        f.setSize(300, 200);
        f.setVisible(true);
    }
}
```

AWT Components

The Interactive Elements

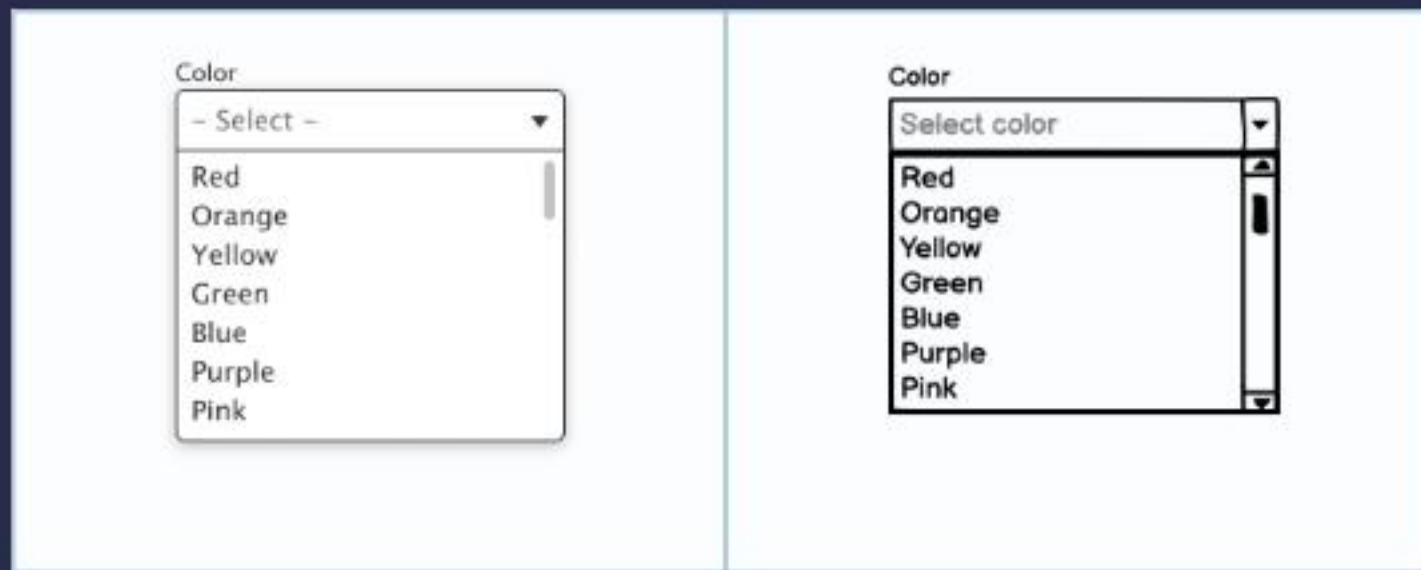# Button Class Program

What is a Button?

The `Button` class creates a simple, labeled push-button.

When clicked, it generates an `ActionEvent`, which your program can "listen" for to perform an action. It's one of the most common GUI controls.

Simple Code:

```
Button b = new Button("Submit");
// Add the button to a Frame or Panel
f.add(b);
```

Choice Class Program

What is a Choice?

The `Choice` class provides a pop-up or drop-down menu of items.

Only one item can be selected at a time. It's used for selecting one option from a list. In Swing, this is replaced by `JComboBox`.

Simple Code:

```
Choice c = new Choice();
c.add("Item 1");
c.add("Item 2");
c.add("Item 3");
f.add(c);
```
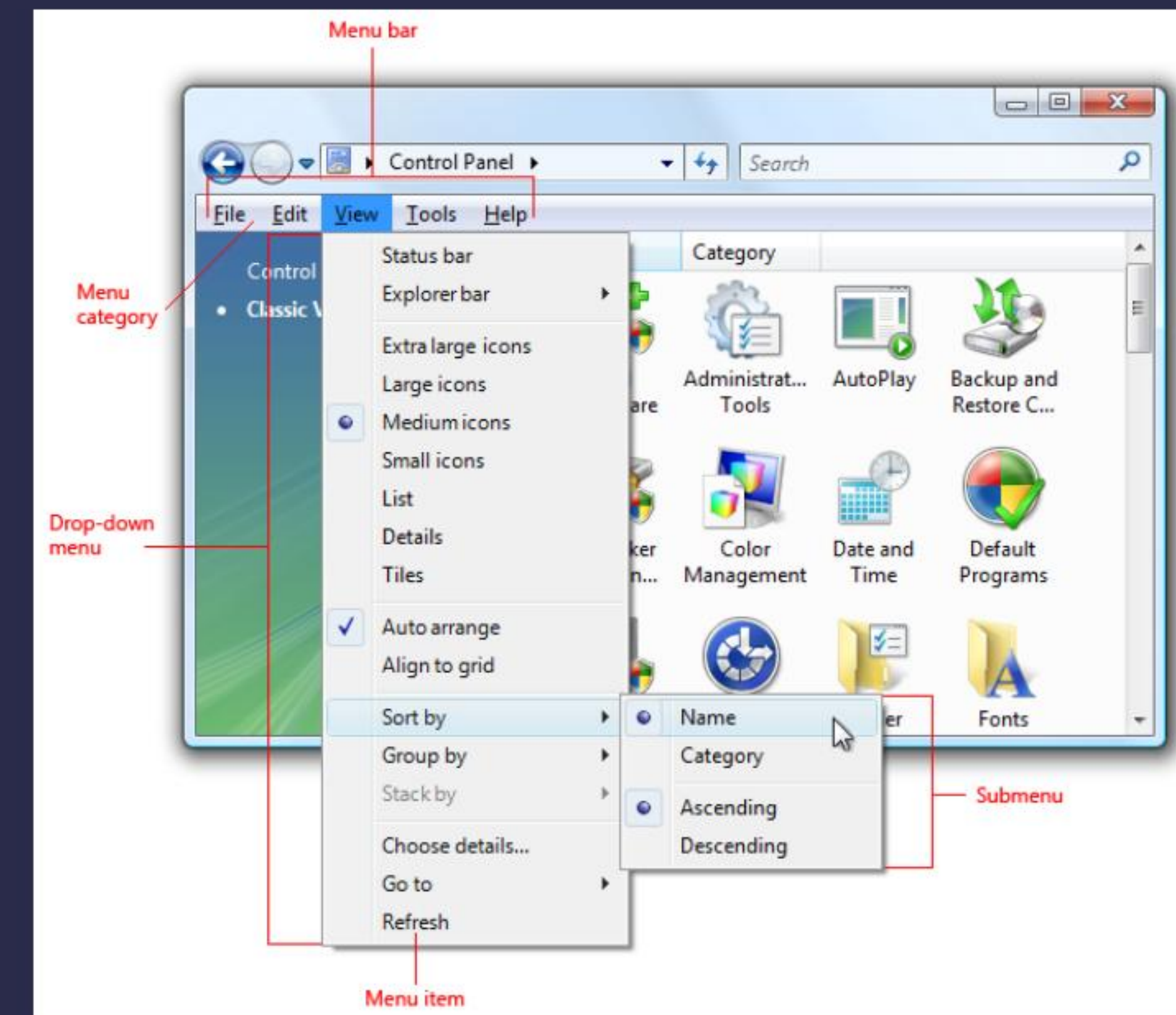
MenuBar Class Program

What is a MenuBar?

This class encapsulates the platform's concept of a top-level menu bar.

- **`MenuBar`:** The bar itself, which is added to a `Frame`.
- **`Menu`:** A single drop-down on the bar (e.g., "File", "Edit").
- **`MenuItem`:** A clickable item within a `Menu` (e.g., "Open", "Save").

Hierarchy: `Frame` → `MenuBar` → `Menu` → `MenuItem`

Summary & Modern Alternatives

AWT Summary

AWT provides the basic, "heavyweight" (OS-native) building blocks of a Java GUI. `Frame` and `Panel` are containers; `Button`,`Choice`, etc., are components.

Java Swing

Swing is the *modern* successor to AWT. It's "lightweight" (painted by Java, not the OS) and has more features and a pluggable look-and-feel (e.g., `JFrame`, `JPanel`, `JButton`).

JavaFX

An even newer, richer framework for building modern desktop applications with advanced media, web, and 3D capabilities. It is the recommended path for new desktop apps in Java.
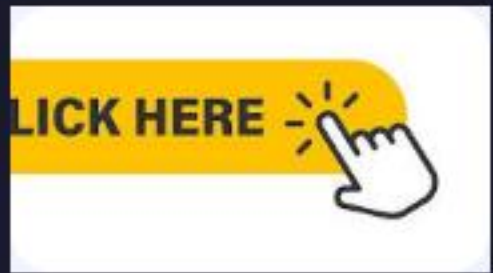
Questions?

Thank you for your attention.

Image Sources

https://cdn.hugeicons.com/blog/1715204120759-UI:UX%20Icon%20libraries.png
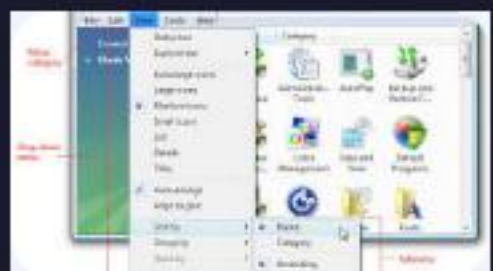
Source: hugeicons.com

https://img.freepik.com/premium-vector/click-here-icon-flat-style-pointer-clicking-vector-illustration-isolated-background-web-button-sign-business-concept_157943-1596.jpg

Source: www.freepik.com

https://balsamiq.com/assets/learn/controls/dropdown-menus/State-open.png

Source: balsamiq.com

https://learn.microsoft.com/en-us/windows/win32/uxguide/images/cmd-menus-image1.png

Source: learn.microsoft.com