

```

use egui::Context;

use crate::{mode::EditMode, GraphEditorApp};

pub fn draw_edit_menu(app: &mut GraphEditorApp, ctx: &Context)
{ egui::SidePanel::left("Menu") .min_width(200.0) .show(ctx, |ui| { app.hovered_on_menu_window =
  ui.rect_contains_pointer(ui.max_rect());

  egui::Frame::new() .inner_margin(egui::Margin::same(10)) .show(ui, |ui| { ui.vertical(|ui|
  { ui.label( egui::RichText::new("Edit Mode").size(app.config.menu_font_size_mini), );

  ui.radio_value( &mut app.edit_mode, EditMode::default_normal(), egui::RichText::new("Normal
  [Esc]") .size(app.config.menu_font_size_normal), ); ui.radio_value( &mut app.edit_mode,
  EditMode::default_add_vertex(), egui::RichText::new("Add Vertex
  [V]") .size(app.config.menu_font_size_normal), ); ui.radio_value( &mut app.edit_mode,
  EditMode::default_add_edge(), egui::RichText::new("Add Edge
  [E]") .size(app.config.menu_font_size_normal), ); ui.radio_value( &mut app.edit_mode,
  EditMode::default_delete(), egui::RichText::new("Delete
  [D]") .size(app.config.menu_font_size_normal), );

  ui.separator();

  ui.label( egui::RichText::new("Indexing [1]") .size(app.config.menu_font_size_mini), );

  ui.radio_value( &mut app.zero_indexed, true, egui::RichText::new("0-
  indexed").size(app.config.menu_font_size_normal), ); ui.radio_value( &mut app.zero_indexed, false,
  egui::RichText::new("1-indexed").size(app.config.menu_font_size_normal), );

  ui.separator();

  ui.label( egui::RichText::new("Direction [Shift + D]") .size(app.config.menu_font_size_mini), );
  ui.radio_value( &mut app.graph.is_directed, false,
  egui::RichText::new("Undirected") .size(app.config.menu_font_size_normal), ); ui.radio_value( &mut
  app.graph.is_directed, true,
  egui::RichText::new("Directed").size(app.config.menu_font_size_normal), );

  ui.separator();

  ui.checkbox( &mut app.is_animated, egui::RichText::new("Animate
  [A]") .size(app.config.menu_font_size_normal), );

  ui.separator();

  if ui .button( egui::RichText::new("Clear All") .size(app.config.menu_font_size_normal), ) .clicked()
  { app.graph.clear(); app.next_z_index = 0; } } } } } }

```