TEMA 07 Uso avanzado de componentes

Los componentes pueden ser mucho mas dinamicos que el simple paso de atributos y manipulacion de sus propiedades , tambien podemos hacer uso intensivo de los templates , uso de Layouts , paso de Contenido completo a un componente

De esta manera se puede estructurar una app de una manera mas reusable y modular

* Templates:
  + Las marcas template indican trozos de codigo html que podemos agrupar incorporando todas las directivas y contenido necesario
* Slots:
  + Los Slots son espacios reservados dentro de la plantilla de un componente donde se puede inyectar contenido. Permiten pasar contenido a un componente Layout desde otro componente, lo que lo hace altamente personalizable y reutilizable
  + Son marcas que se incorporan con una directiva v-slot donde se indica que en ese lugar se “incrusta” todo el contenido de otro componente , muy util para definicion de bloques

// titled-frame.vue

<template>

<div class="frame">

<header><h2><slot name="header">Title</slot></h2></header>

<slot>This is the default content if nothing gets specified to go here</slot>

</div>

</template>

// app.vue

<template>

<titled-frame>

<template v-slot:header>

<!-- The code below goes into the header slot -->

My Image’s Title

</template>

<!-- The code below goes into the default slot -->

<img src="an-image.jpg">

</titled-frame>

</template>

* Layouts:
  + Es un sistema de organización de los componentes , lo haremos a modo de carpetas incorporando nuestros compeontes
  + Los Layouts en Vue te permiten definir la estructura de un componente mientras dejas áreas específicas abiertas para la personalización. Esto es particularmente útil para crear diseños de página consistentes y al mismo tiempo permitir variaciones de contenido.
* Componentes Dinamicos:
  + Incorporamos en la marca el componente referenciado por :is , si la variable cambia , cambia el contenido , ahí su potencial
  + <component :is="componentTagName"></component>

Ejercicio:

* Creamos nuestra primera app grafica y multi compoente:
* Aprovechamos para instalarnos vuetify
* Montaremos una barra superior basica donde pondremos varios botones para redireccionar a distintos componentes
  + Home
  + Login
  + Register
* Cada componente es diferente e incialmente minimos
* Creamos una carpeta en componentes llamada layout
  + Src/components/MainLayout.vue
  + Src/components/LoginLayout.vue
  + Src/components/RegisterLayout.vue
* Retocamos App.vue
  + Dejamos la barra superior e incorporamos unos botones
  + ¡Importante! Creamos una data/var llamada layout para referenciar a nuestro componente
  + Ponemos algun boton para cambiar de componente