

PRIMER EXAMEN PARCIAL

02/05/2022

😊 Leé por lo menos dos veces el enunciado antes de resolver.

## Enunciado

Se te encomienda el desafío de proponer un prototipo para la nueva biblioteca de componentes de interfaces gráficas de usuario (UI) para el nuevo smart Ort Phone que se desarrollará en un lenguaje orientado a objetos.

Ort Phone nos pide que todos los componentes si o si tengan un método dibujar() es decir que sean dibujables y nos encomienda modelar los siguientes componentes con las características descritas a continuación:

**Botón:** los cuales tienen un texto, podrá tener el color que se quiera definir para el mismo, se les debería poder asignar un determinado ancho como también su altura y se debe permitir ubicarlos en cualquier ubicación dentro de los ejes X e Y. Todo botón podrá estar habilitado o inhabilitado.

**Campo de Texto:** se les podrá ingresar o setear un texto, debe poder tener la posibilidad de ingresar un párrafo de texto (multitexto), se debe permitir ubicarlos en cualquier ubicación dentro de los ejes X e Y, el fondo del mismo podrá tener el color que se quiera definir, se les debería poder asignar un determinado ancho como también la altura. Todo campo de texto podrá estar habilitado o inhabilitado y si queremos que el cursor se sitúe sobre el mismo deberíamos poder indicar dicha acción.

**Etiqueta:** podrán mostrar un texto, ese texto podría ser con una fuente determinada y podrá tener el color que se quiera definir para el mismo, se les debería poder asignar un determinado ancho como también su altura y se debe permitir ubicarlos en cualquier ubicación dentro de los ejes X e Y. Toda etiqueta de texto podrá estar habilitada o inhabilitada.

**Ventana:** las ventanas tienen la particularidad que podrá tener otros componentes dentro de la misma y podrá tener 4 estados posibles (MINIMIZADA, MAXIMIZADA, CERRADA, ABIERTA), deben poder tener un texto que se mostrará como título en la barra superior, podrá tener cualquier color de fondo que se quiera asignar a la misma, se les debería poder definir un determinado ancho como también su altura y se debe permitir ubicarla dentro de los ejes X e Y de la pantalla. Toda ventana a su vez podrá estar habilitada o inhabilitada como cualquier otro componente ya que una ventana es también un componente.

**Ventana de Error:** es una ventana con las mismas características y comportamiento que una Ventana pero que ya tiene una etiqueta que mostrará un mensaje de error y un botón con el texto "Aceptar".

Basado en el enunciado descripto, realizar:

A) El diagrama de clases que lo modelice, con sus relaciones, atributos y métodos.

B) El método **dibujar()** que debe **mostrar por consola** los siguientes datos:

1. En el caso de un botón: "Dibujando Botón con el texto: XXX con un alto de: XXXmm y ancho de: XXXmm".
2. En el caso de un campo de texto: "Dibujando Campo de Texto XXX que tiene el cursor ZZZ (donde XXX puede ser "multitexto" o "simple" y ZZZ "activado" o "desactivado")"
3. En el caso de una etiqueta: "Dibujando Etiqueta con el texto XXX con fuente XXX"

PRIMER EXAMEN PARCIAL

02/05/2022

4. En el caso de cualquier ventana: "Dibujando Ventana con los siguientes componentes: " y a continuación se deberán dibujar cada componente que tenga dentro.  
(en el caso de que ninguno de los componentes de Campo de Texto que se agregaron a la ventana tenga un cursor activado, activar el primer Campo de Texto que fue agregado)
- C) El método **agregar** de la clase Ventana que debe permitir agregar sobre la misma cualquier componente excepto si es una Ventana. En caso de que el componente se pueda agregar devolver true caso contrario si es una Ventana devolver false.
- D) Un **constructor** con parámetros para la VentanaDeError que permita ingresar el texto de error.

## Criterios

Para considerar aprobado el trabajo práctico, el mismo debe demostrar la correcta aplicación de los siguientes conceptos de la programación orientada a objetos:

- Correcta definición de clases y asignación adecuada de sus responsabilidades.
- Encapsulamiento, ocultamiento de información y uso de getters y setters sólo cuando corresponda.
- Modularización reutilizable y mantenible con uso de métodos con correcta parametrización.
- Correcta aplicación de miembros de instancia y de clase.
- Correcta aplicación de herencia y polimorfismo.
- Correcta aplicación conceptual de las relaciones entre clases.

% Correcto	0 a 20	25 a 45	50 a 55	60	65 a 70	75	80	85 a 90	95	100
Nota	1	2	¿4?	4	5	6	7	8	9	10