Modificaciones

Clase App:

* restaurante: Restaurante 🡪 Se utilizó para tener una variable con la cual utilizar los métodos de esta clase, es la principal unión entre lo que ve el usuario y la lógica detrás
* ejecutarAñadirPedido() 🡪 Se utilizó para añadir productos al pedido, así como también para modificarlos añadiendo ingredientes u otros productos extras
* ejecutarCargarTodo() 🡪 Se utilizó para cargar todos los archivos .txt
* ejecutarInformacionCliente() 🡪 Se utilizó para crear un nuevo objeto Pedido con la información que entra el usuario sobre su nombre y su dirección
* input(String) 🡪 Se utilizó para simplificar el modo de ingresar un dato que da el usuario

Clase Restaurante:

* elPedidoActual 🡪 Atributo utilizado para saber qué objeto de Pedido se está utilizando y acceder a métodos de este para añadir o eliminar información
* ajustadoActual 🡪 Atributo utilizado para acceder al objeto de ProductoAjustado y así editarlo dependiendo de qué se necesitaba, añadir o eliminar un ingrediente, etc.
* ListaMenu 🡪 Atributo utilizado para guardar los objetos que se generan a partir del archivo menu.txt
* Ingredientes 🡪 Atributo utilizado para guardar los objetos que se generan a partir del archivo ingredientes.txt
* ListaCombos 🡪 Atributo utilizado para guardar los objetos que se generan a partir del archivo combos.txt
* buscarIngrediente() 🡪 Método utilizado para observar si ya existe el producto dentro de la lista Ingredientes
* crearProductoAjustado() 🡪 Método para crear un objeto de un ProductoAjustado
* añadirIngProdAjus() 🡪 Añadir un nuevo ingrediente al objeto de ProductoAjustado
* eliminarIngProdAjus() 🡪 Eliminar un ingrediente al objeto de ProductoAjustado
* getProductoAjustadoActual 🡪 Devuelve el objeto de la clase ProductoAjustado que ese está usando actualmente
* agregarItemaCombo 🡪 Poder acceder a agregarItem () del objeto Combo, para añadir un producto al objeto de Combo
* getMenuProductos() 🡪 Mostrar todos los productos y combos disponibles
* getMenuIngredientes 🡪 Mostrar todos los ingredientes disponibles

Clase Pedido

* prodPedido List<Producto> 🡪 Agregar nuevos objetos de la clase Producto a una lista que contenga todos los productos del pedido

Clase Combo

* Combo(String, double, String, String, String, int) 🡪 Se cambió la entrada de los atributos para el método, original recibe dos, el mío recibe 5. Se requería guardar los productos que se incluían dentro del combo, Dado que los atributos son características del objeto , no se pueden olvidar estas características, se deben incluir

Clase ProductoAjustado

* Base: ProductoMenu 🡪 Se debe utilizar un objeto que contenga el ProductoBase de productoMenu que se va a modificar
* agregados: ArrayList<Ingrediente> 🡪 Se crea una lista para guardar todos los ingredientes añadidos al producto
* eliminados: ArrayList<Ingrediente> 🡪 🡪 Se crea una lista para guardar todos los ingredientes eliminados del producto