I fråga 2&3 skall dessa punkter skrivas ner och lämnas in i ett separat dokument.

1.Identifiera vad som ska göras och vilken information som ska behandlas

A) Vad skall programmet göra? Agera som en meny.

Programmet ska fungera som en digital meny där gästen kan välja mellan 4 olika rätter, 4 olika drycker och fyra olika efterrätter. De här 3 olika kategorierna bör vara huvudklasser så att man lätt kan skapa fyra olika sorter med liknande huvudattribut. När gästen har gjort sina val, ska programmet räkna ut totalpriset för beställningen (ett av attributen bör pris!) och visa en vad användaren har valt. Här ska gästen kunna ändra sina val och avbryta beställningen (Innan en betalning har skett). Programmet skall vara lätt använt så att kunden förstår vad som händer (GUI), det skall också vara snabbt och smidigt.

B) Vad finns på nätet?

Ett fler tal matbeställnings appar /hemsidor.

Mallar till hur man bör bygga upp en sådan product.

Hemsidor för att hjälpa lösa problem längst vägen så som: ChatGPT, stackoverflow, forums, youtube mm.

Python kod och hur det fungerar.

Lucidchart för att rita upp funktionerna och vad programmet skall göra.

<https://github.com/TomSchimansky/CustomTkinter/wiki>

<https://github.com/TomSchimansky/CustomTkinter>

C) Olika funktioner i programmet

Klasser:  
Food, Drink, Dessert, Order, App, CTkOptionMenu

Food attribute: name, price

Drink attribute: name, price

Dessert attribute: name, price

D) Olika kodexempel

2.Bryt upp problemet i mindre delar och börja skrivahalvkod

3.Tills alla delar av projektet är kodade:

A) Koda en mindre del så att den fungerar med andra fungerande delar

B) Vid behov skriv om halvkod och kod så att programmet blir bättre

C) Skriv dokumentation och kommentarer för fungerande kod

CustomTkinter-baserad GUI-applikation för att beställa mat. Den innehåller tre klasser, Food, Drink och Dessert, var och en med egenskaperna namn och pris. Därefter finns det tre listor med mat, drycker och efterrätter, som används för att skapa dropdown-menyer i GUI:n.

Den huvudsakliga App-klassen ärver från en customtkinter-klass (som importerats) och använder dess ”widgets” (inbyggda funktioner i costumTkinter) för att skapa gränssnittet. Det finns tre dropdown-menyer, en för mat, en för drycker och en för efterrätter, samt en callback-metod för varje meny som skriver ut valet när en användare gör ett val.

Alternativ dokumentation.

I denna kod skapas en GUI-applikation med hjälp av biblioteket tkinter och en customtkinter-bibliotek. Programmet innehåller följande klasser och objekt:

Klassen Food: Denna klass representerar matvaror och har två attribut, namn och pris.

Klassen Drink: Denna klass representerar drycker och har två attribut, namn och pris.

Klassen Dessert: Denna klass representerar efterrätter och har två attribut, namn och pris.

Listorna foods, drinks och desserts: Dessa listor innehåller objekt av Food, Drink och Dessert-klassen, respektive.

Klassen CTkOptionMenu: Denna klass är en subklass av tkinter.OptionMenu och tillåter oss att hämta det valda alternativet från en dropdown-meny.

Klassen App: Denna klass är en subklass av customtkinter.CTk och representerar vår huvudapplikation. I denna klass definieras layouten och funktionerna för vår applikation, inklusive:

FoodOM, DrinkOM och DessertOM: Dessa är instanser av CTkOptionMenu som visar mat, dryck och efterrätter.

textbox: Detta är en instans av customtkinter.CTkTextbox som visar det valda alternativet.

optionmenuCallback(): Denna metod uppdaterar textbox med det valda alternativet från en av dropdown-menyerna.

buttonClick(): Denna metod skriver ut en loggmeddelande när en knapp trycks.

Sammantaget är denna kod en enkel applikation som tillåter en användare att välja mat,

D) Dokumentera vad som finns kvar att göra.

4.Gå igenom koden och se till att onödig kod, onödiga kommentarer och onödig dokumentation tas bort.

5.Om tid finns gå igenom koden och se om den går att optimera. Glöm inte arbetsgång med länkar och kodavsnitt