

Kassa

en enkel kassa mjuvvara

Sam Sadeghi SY17

2019/2020

Abstract

Innehåll

1	Inledning	3
1.1	Syfte	3
1.2	Mål	3
1.3	Metod	3
2	Utförande	5
3	Resultat	8
4	Diskussion	8
5	Slutsats	8

1 Inledning

Ibland hjälper jag min farbror i hans livsmedel butik. Där får jag använd använda mig av kassaapparaten. Programmet som kassaapparaten på farbrors butik kör är långsamt och jag tänkte att jag skulle kunna göra ett liknande program som kör snabbare genom att endast ha med de nödvändigaste funktionerna.

1.1 Syfte

I skapandet av det här programmet ska jag lära mig hur man gör ett grafisk program med hjälp av programmerings språket Python och gränssnittsverktyget Tkinter.

1.2 Mål

Vid slutet av arbetet ska jag ha ett enkelt kassa program som kan räkna ihop summan av flera varor samt hur mycket moms som skall betala beroende på varans typ. Programmet skall även kunna skriva ut ett kvitto

1.3 Metod

Redskap jag kommer använda är

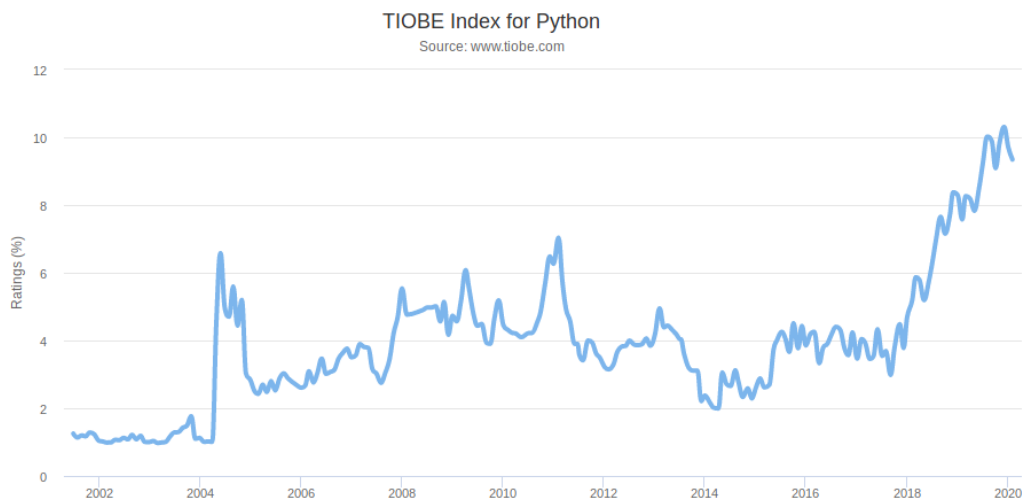
Vim:

Vim är ett text redigerings program som körs i konsolen. Det finns massor med fördelar med Vim. En av dessa fördelar är att den inte tar lika mycket resurser att köra Vim som till exempel Visual Studio men det i sig är inte så viktigt utan den största anledningen till varför jag jag använde Vim för att skriva det här programmet är att jag har vant mig vid Vims massor med tangentbords-genvägar och det hjälper mig skriva snabbare inte bara det utan jag är också väldigt bekväm med att konfigurera Vim så att lättare kan skriva. Ett av dessa konfigurationer jag har gjort för att underlätta mitt skapande av programmet är att jag har gjort så att när jag trycker ner på **Ctrl** och **W** samtidigt så startar den programmet. Andra program kan säkert göra någonting liknade eller kanske till och med kommer inbyggt med en sådan funktion men i Vim får jag göra det det själv som får mig att bättre förstå vad som händer i bakgrunden och som sagt tidigare är jag mycket bekvämare med Vim än vad jag är med andra text redigerings program

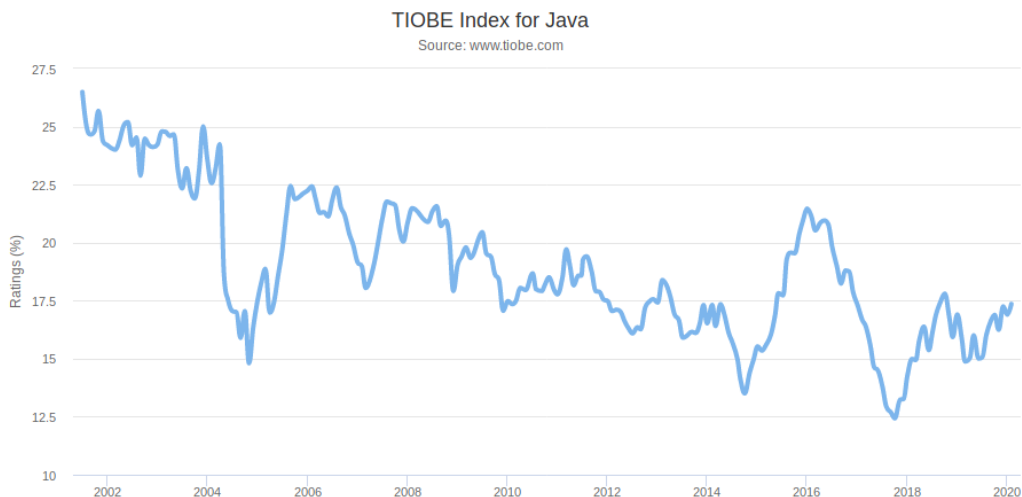
Python:

Python är ett programmeringsspråk. Jag valde just Python för att skriva det

här programmet för att jag ser det väldigt ofta i min omgivning. Enligt TIOBE indexet [2] är Python det tredje populäraste språket och dess popularitet värkar öka (se figur 1) till skillnad från språket som ligger på första plats det vill säga Java (se figur 2) och ju populärare ett programmeringsspråk är så finns det fler projekt som är skrivna i de programmeringsspråket därför tycker jag att det är värt att lära sig /skriva i Python.



Figur 1: TIOBE index för python



Figur 2: TIOBE index för java

Tkinter/TK:

Tkinter är en API för att använda TK i Python. TK är grafiskt gränssnitt

verktyg som fungerar på Windows, MacOS och Linux. Jag valde Tkinter på grund av att den fungerar på dom tre vanligare operativsystemen. Tkinter kommer

FPDF:

FPDF är ett programbibliotek för att generera PDF filer. PDF filer är dokument filer som är optimerade för att skrivas ut.

Github och Git:

Github är en webserver med fokus på program utveckling. Github använder sig av version kontroll programmet Git. Git gör det enkelt för mig att se vad jag har gjort och när jag har gjort det. Git tillsammans med Github låter mig jobba på programmet var jag än är utan att vara uppkopplad till internet

Tutorialpoints sida på Python GUI Programming:

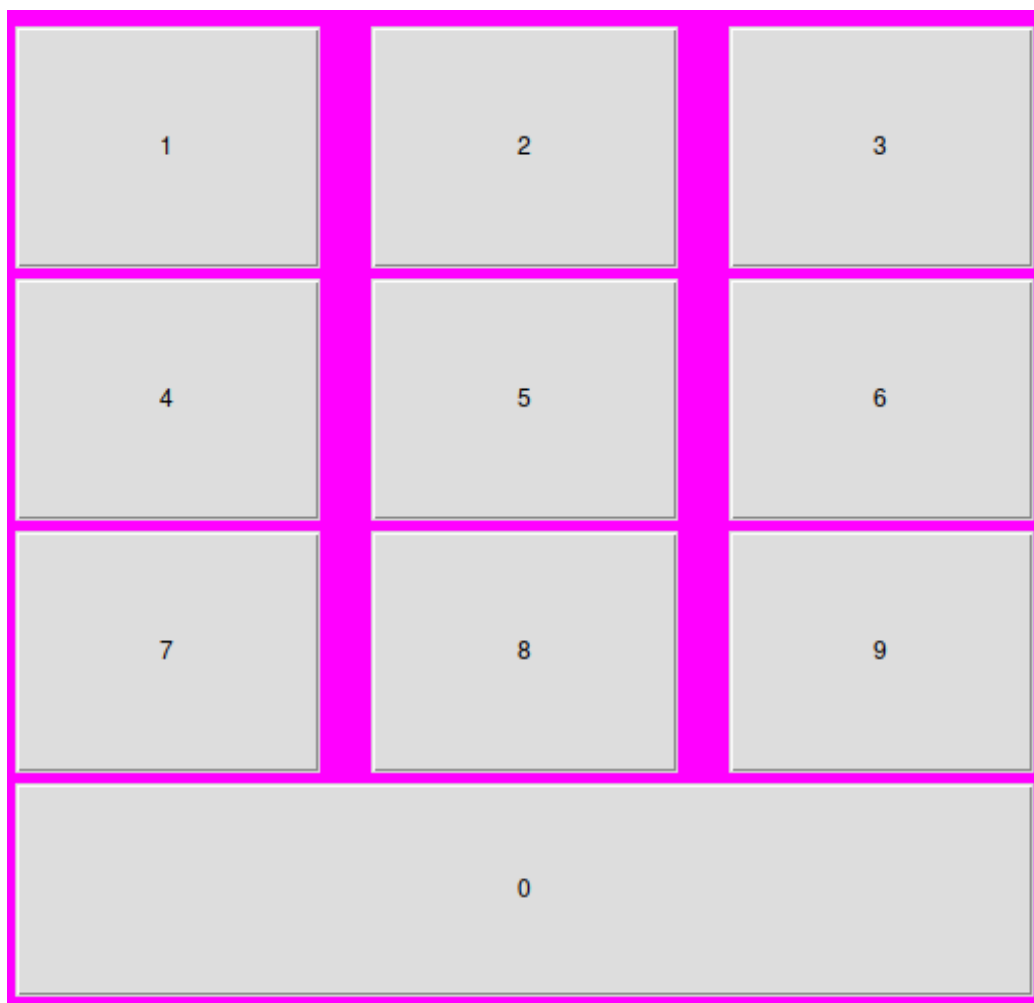
Tutorialpoints sida på Python GUI Programming är en webbsida som har i syfte att lära hur man använder TKinter. Jag valde att använda Tutorialpoints sida istället för den officiella TKinter manualen efter som Tutorialpoints sida är mycket enklare att hitta information på. Det finns en lista på alla Widgets och en kort beskrivning av vad dem används för; du kan klicka på namnet av en widget du vill veta mer om och du tas till en sida där du får veta syntaxen och alla tillval för widgeten du klickat på.

2 Utförande

När jag börja projektet hade jag aldrig tidigare skrivit program i Python och det ända grafiska gränssnit jag tidigare använt var HTML på Webben så det första jag behövde göra var att lära mig. Youtube videon "How to Program a GUI Application (with Python Tkinter)!" av Keith Galli[1] användes för att få en förståelse av hur man gör grafiska program med Tkinter och python.

Först behövdes ett sätt att skriva in siffror i programmet. Man skulle kunna använda nummer knapparna på tangentbordet men eftersom ska kunna användas med endast en pekskärm så behövdes nummer knappar på skärmen, Detta gjorde jag genom att skapa tio knappar (från 0 till 9) och placera dom i en 3 gånger 3 rutnät med en stor knap för noll längst ner (se figur 3).

Därefter behövdes det ett sätt att se vad för nummer man skrivit; för det användes det en lable som döptes till text tillsammans med en global variabel som deklarerades med namnet num. Num är noll när programmet startar men när en nummer knapp trycks så körs en funktion vid namnet knapp_tryck med värdet på knappen; funktionen kommer sedan att multiplicera num med tio, addera värdet den fick in till num och tillslut visa det nya värdet för num på text. Nu kan användaren skriva in siffror i programmet men det saknas



Figur 3: hur nummer knapparna ser ut

ett sätt att ta bort eller ångra siffror man skrivit därför implementerades till en knapp som raderar den sista siffran på talet och en knapp som tar bort hela talet. Knappen som tar bort sista siffran är märkt med symbolen "<" och fungerar genom att kalla på funktionen `knapp_del`. Funktionen `knapp_del` fungerar på så sätt att den konverterar num till en sträng (detta görs för att det är lättare att kolla storleken och ta bort sista karaktären av en sträng). När num har konverterats till en sträng så kontrolleras det om num har fler än en karaktär i så fall tas den sista karaktären bort; om num bara har en karaktär kommer num sättas till noll. Knappen som tar bort hela talet har bokstaven "C" på sig och kallar på funktionen `knapp_clear` när den är tryckt. `Knapp_clear` är en funktion som byter värdet av num till noll.

Nu när det finns nummer knappar och ett sätt att se vilket nummer man har skrivit så behövs det ett sätt att säga till programmet vad numret betyder. Numret som har skrivits in kan vara en av tre saker. Det kan vara en produkt, en betalning eller ett antal.

En produkt är något som kunden ska köpas. Produkter representeras i koden med klassen prod. Prod har attributen namn, antal, pris och moms och en funktion som heter totpris som ger priset multiplicerat med antalet. Man väljer att numret är en produkt genom att trycka på en av produkt knapparna (se figur 4) efter att skrivit in ett nummer. Numret blir på priset av produkten.

En betalning är hur kunden betalar för produkten. Betalningar behandlas av koden likt hur produkter behandlas det vill säga betalningar representeras av klassen betal som är precis som klassen prod med skillnaden att betal inte har moms attributen. Man säger till programmet att numret är en betalning genom att trycka på en av betalnings knapparna (se figur 4) efter man har valt ett nummer. Numret blir hur mycket som ska betalas med valt betalnings alternativ. Om numret är noll kommer knappen att bli grön när och när man trycker på en grön knapp så lägger programmet en betalning som är lika stort som total priset för alla produkter minus alla tidigare betalningar.

Antal är som sagt tidigare en attribut hos produkter och betalningar. Antal för den senaste produkten eller betslingen kan ändras men plus och minus knapparna (se figur 4). När man trycker på en av antals knapparna så kollar programmet om num är noll; ifall num är noll så lägger den till eller tar bort en av antalet på den senaste

+	-	ånga	livsmedel	godis
kort	kontant		hygien	dricka

Figur 4: Till höger är dom fyra produkt knapparna markerade som livsmedel, godis, hygien och dricka. Till vänster är dom två betalnings knapparna markerade med orden kontant och kort. Ovanför betalnings knapparna är antals knapparna märkta med "+" och "-".

3 Resultat

4 Diskussion

5 Slutsats

Referenser

- [1] Keith Galli. *How to Program a GUI Application (with Python Tkinter)!* Youtube. 2019. URL: <https://www.youtube.com/watch?v=D8-snVfekto>.
- [2] TIOBE. *TIOBE Index for February 2020*. 2020. URL: <https://www.tiobe.com/tiobe-index/> (hämtad 2020-03-01).