

Benvenuto Alla Radiant Cow Steak House !
Welcome aboard.



Di cosa si tratta?

La Radiant Steakhouse é un progetto fatto con un unico scopo, gestire un improbabile e imbarazzante steakhouse, attraverso un programma scritto in linguaggio Python in co-operativa tra 3 grandi aziende:

Save Michael Enterprise (Azienda no profit dedicata alla carit )

Frankes Big Farm Metroin (Azienda governativa)

Saint Cory Co. (Azienda produttrice di armi di distruzione di massa.)

Com'è strutturato il programma ?

Il programma (Scritto in linguaggio Python e C#) é organizzato in 4 parti principali (Main, 3 procedure, con circa 200 righe di codice

2 - Menu Clienti



3 - Funzione di associazione tavolo



4 - Selezione di ruolo



5 - Generazione codice lavoratore



5 - Main



```
3 > def MenuClienti(): ...  
126  
127 > def tavoli(): ...  
129  
130  
131  
132 > def selruolo(): ...  
142  
143 > def codice(): ...  
145  
146 > def main(): ...
```

Incominciamo dalla portata principale. (Capita? Eh?)

Il Main() é effettivamente la parte piú grande del codice in quanto ha lo scopo di richiamare a sé varie funzioni ed eseguire istruzioni.

Il codice va a dividersi effettivamente in 2 parti principali tramite l'uso della funzione SelRuolo()

```
146 def main():
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198

    while True:

        n = RoleSelect()

        if n == '1':
            Number = Tables()
            print("Welcome to Radiant Cow Steak House's customer service.")
            print("")
            print(f"Please take your seat at table N: {Number}")
            print("Here we go! Here you have the menu, take your time and make yourself home.")
            CustomerMenu()

        if n == '2':
            while True:
                print("Hello! Have I seen you anywhere else before? (1 - Yes, 2 - No)")
                loR = input()

                if loR == "2":
                    print("Ah that's right! First time around ? What's your name: ")
                    name = input()
                    print(f"(name), eh? Beautiful name! How old are you?")
                    age = input()
                    print(f"So you are {age}, huh?")
                    print("")
                    print("Well now you'll receive your code, please note it down and don't lose it. . .")
                    print("")
                    cod = Code()
                    print(f"Here's your code: {cod}")

                if loR == "1":
                    print("Please, insert the code you were given when you first signed in.")
                    while True:
                        insCode = input()
                        if insCode == str(cod):
                            print(f"Ah that's right, you're {name} and you're {age}")
                            workHours = random.randint(1,9)
                            hour = random.randint(0,23)
                            minutes = random.randint(0,59)
                            startHour = hour,minutes
                            endingHour = hour + workHours
                            cTime = endingHour,minutes
                            print(f"Your shift's going to last {workHours} hours starting off at {startHour} and end at {cTime}.")
                            break
                        else:
                            print("Invalid code, try again. . .")

            if n == "3":
                print("Have a good day!")
                break
```



Employee and customer

The function RoleSelect() is what allows the program to function properly, effectively swapping between modes:

1 - Customer mode

2 - Employee mode.

3 - Exit (The program closes)

```
132 def selruolo():
133     print("Benvenuto nel programma, inserire modalità 1 - 'Cliente' oppure 2 - 'Dipendente' oppure 3 - Esci")
134     while True:
135         choice = input()
136
137         if choice == '1' or '2' or '3':
138             return choice
139         else:
140
141             print("Scelta non valida. Per favore, inserire un opzione valida. . .")
```



What does the customer mode fully offer?

Customer mode tells the customer where to seat at, offering them one of the many tables we have, as well as giving them a warm welcoming message along with our fancy menu !

```
153     if n == '1':
154         Number = Tables()
155         print("Welcome to Radiant Cow Steak House's customer service.")
156         print("")
157         print(f>Please take your seat at table N: {Number}")
158         print("There we go! Here you have the menu, take your time and make yourself home.")
159         CustomerMenu()
160
```

The menu !

Right after the customer is welcomed they are given a various set of food options to select and consume at their table.

With the press of a button a dish is added to their check and they are even given the chance of receiving a free coffee !

```
3 def CustomerMenu():
4     print("\nWelcome to the Radiant Cow Steak House!")
5     print("-----")
6     print("|                MAIN MENU                |")
7     print("-----")
8     print("|    APPETIZERS                            |")
9     print("| 1. Beef Carpaccio           €15.00      |")
10    print("| 2. Beef Tartare            €16.00      |")
11    print("| 3. Mixed Bruschetta         €8.00       |")
12    print("-----")
13    print("|    MAIN COURSES                       |")
14    print("| 4. Tagliatelle with Ragù    €14.00     |")
15    print("| 5. Mushroom Risotto        €13.00     |")
16    print("| 6. Gnocchi with Gorgonzola €12.00     |")
17    print("-----")
18    print("|    MEAT DISHES                        |")
19    print("| 7. Ribeye Steak (300g)      €28.00     |")
20    print("| 8. Filet (250g)             €32.00     |")
21    print("| 9. T-Bone (500g)           €35.00     |")
22    print("| 10. Tomahawk (800g)         €45.00     |")
23    print("-----")
24    print("|    SIDES                             |")
25    print("| 11. Roasted Potatoes        €5.00      |")
26    print("| 12. Grilled Vegetables      €6.00      |")
27    print("| 13. Mixed Salad             €4.00      |")
28    print("-----")
29    print("|    DRINKS                             |")
30    print("| 14. House Wine              €18.00     |")
31    print("| 15. Craft Beer              €6.00      |")
32    print("| 16. Mineral Water           €2.50      |")
33    print("-----")
34    print("|    DESSERTS                         |")
35    print("| 17. Tiramisu                €6.00      |")
36    print("| 18. Panna Cotta             €5.00      |")
37    print("| 19. Cheesecake              €6.00      |")
38    print("-----")
39    print("|    AFTER MEAL                      |")
40    print("| -1- Coffee                  €1.00      |")
41    print("| -2- Digestive               €4.00      |")
42    print("-----")
```

Employee work

As a second option at the beginning of the program, we have the employee mode, which in turn offers us 2 distinct and separate options: one for registering the worker and another for viewing.

```
161 if n == '2':
162     while True:
163         print("Ciao! Ti ho già visto da qualche parte? (1 - Sì, 2 - No)")
164         lOrR = input()
165
166         if lOrR == "2":
167             print("Ah, ecco! Mi pareva di non averti mai visto in giro... sei nuovo? Come ti chiami?")
168             nome = input()
169             print(f"{nome}, eh? Bellissimo nome! Quanti anni hai?")
170             età = input()
171             print(f"Quindi hai {età} anni, eh? Te li porti benissimo.")
172             print("")
173             print("Bhe ora ti darò il tuo codice ti chiederei di non perderlo")
174             print("")
175             cod = codice()
176             print(f"Ecco il tuo codice {cod}")
177
```

```
161 if n == '2':
162     while True:
163         print("Ciao! Ti ho già visto da qualche parte? (1 - Sì, 2 - No)")
164         lOrR = input()
165
166         if lOrR == "2":
167             print("Ah, ecco! Mi pareva di non averti mai visto in giro... sei nuovo? Come ti chiami?")
168             nome = input()
169             print(f"{nome}, eh? Bellissimo nome! Quanti anni hai?")
170             età = input()
171             print(f"Quindi hai {età} anni, eh? Te li porti benissimo.")
172             print("")
173             print("Bhe ora ti darò il tuo codice ti chiederei di non perderlo")
174             print("")
175             cod = codice()
176             print(f"Ecco il tuo codice {cod}")
177
```




We're almost done !



```
127 def Tables():
128     return random.randint(1, 21)
129
130
131
132 > def RoleSelect():...
142
143 def Code():
144     cod = random.randint(1000000, 99999999)
145
146 def main():
147
148
149     while True:
150
151         n = RoleSelect()
152
153 >     if n == '1':...
160
161 >     if n == '2':...
194
195         if n == "3":
196             print("Have a good day !")
197             break
198
199
200 if __name__ == "__main__":
201     main()
```

The program is coming to an end, the last remaining tasks are simple functions used to generate random values and the exit option that allows the program to terminate the program will only end if said option is pressed, otherwise, it will loop back.



The graphical version of the program

While our main project is a boring, console focused program, we also took the initiative to make a much better GUI version



Versione in C# !

Una volta eseguita la versione python abbiamo deciso di farne una anche in C# !

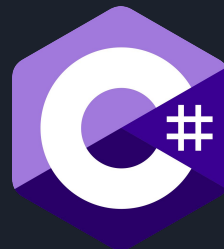
Questo poiché entrambi i linguaggi offrono diversi vantaggi e difetti principali, ma scopriamo quali sono nello specifico...

```
0 riferimenti
1 class Program
2 {
3     static Random rand = new Random();
4
5     1 riferimento
172 static void MenuCliente()...
173
177     1 riferimento
178 static int Tavoli()...
194
195     1 riferimento
199 static string SelezioneRuolo()...
200
201     2 riferimenti
202 static int Codice()...
203
204     0 riferimenti
205 static void Main()
206 {
207     while (true)
208     {
209         string n = SelezioneRuolo();
210
211         if (n == "1")...
212
213         if (n == "2")...
214
215         if (n == "3")
216         {
217             Console.WriteLine("Buona giornata!");
218             break;
219         }
220     }
221 }
```

Il C# E il python hanno alcune differenze, ma quali sono?



- 1 - Linguaggio interpretato.
- 2 - Il Python non richiede di tipizzare le variabili.
- 3 - Un linguaggio diretto, con uno stile più imperativo, che comunque supporta una programmazione ad oggetti.
- 4 - Linguaggio multiplatforma open source, supportato su sistemi operativi Windows, Linux, MacOS.



- 1 - Linguaggio compilato.
- 2 - É necessario il tipo di variabile alla dichiarazione.
- 3 - Linguaggio ideato per la programmazione ad oggetti.
- 4 - Linguaggio usato per lo più in programmi fatti per sistema operativo Windows. (Per quanto con .net 5.0 si possa programmare anche per macchine Linux, MacOS ecc...)



Difficoltà e problemi tecnici:

A causa del periodo in quale questo progetto é stato assegnato ci sono stati moderati problemi di organizzazione tra i partecipanti del gruppo tuttavia questo non ha impedito la progressione del progetto, grazie al capogruppo.

Il Python non fa utilizzo di parentesi graffe per definire le dimensioni dei blocchi di istruzione ma utilizza un sistema basato sulla posizione ci ha causato non pochi giramenti di scatole, in quanto per noi è stata la prima volta.

C# non ci ha dato particolari problematiche per quanto riguarda il codice, ma abbiamo avuto più grattacapi con la “traduzione” da Python a C# date le sostanziali differenze.



Quindi, risultati finali ??

Il programma é stato sviluppato da un team di 3 persone ed ha richiesto un numero di circa 40 ore in tutto, con 13 incontri totali e sotto le seguenti informazioni:

Come é strutturato:

Il programma ha una sintassi semplice, basata sulla ricerca della facilità di utilizzo per l'utente in modo che il programma possa essere compreso da chiunque

abbiamo tentato di creare un ambiente amichevole con una sintassi di testo per il nostro "operatore" gentile e accogliente in modo che sia un piacere venire da noi sia per clienti che per dipendenti

Quanto può effettivamente fruttare alla steak house:

La steak house può risparmiare sia sul personale che la risorsa fisica del menú, e poi il programma permette di operare costantemente senza la presenza di un cameriere.

Ore di lavoro complessivo: 40

Possiamo stimare circa 18.000\$



~THE END~

Capo gruppo: Salvato Michele

Colleghi: Santaniello Gennaro, Matrone Francesco Pio