

UNIVERSIDADE FEDERAL DO PARANÁ

VICTOR LUIZ RODRIGUES SILVA

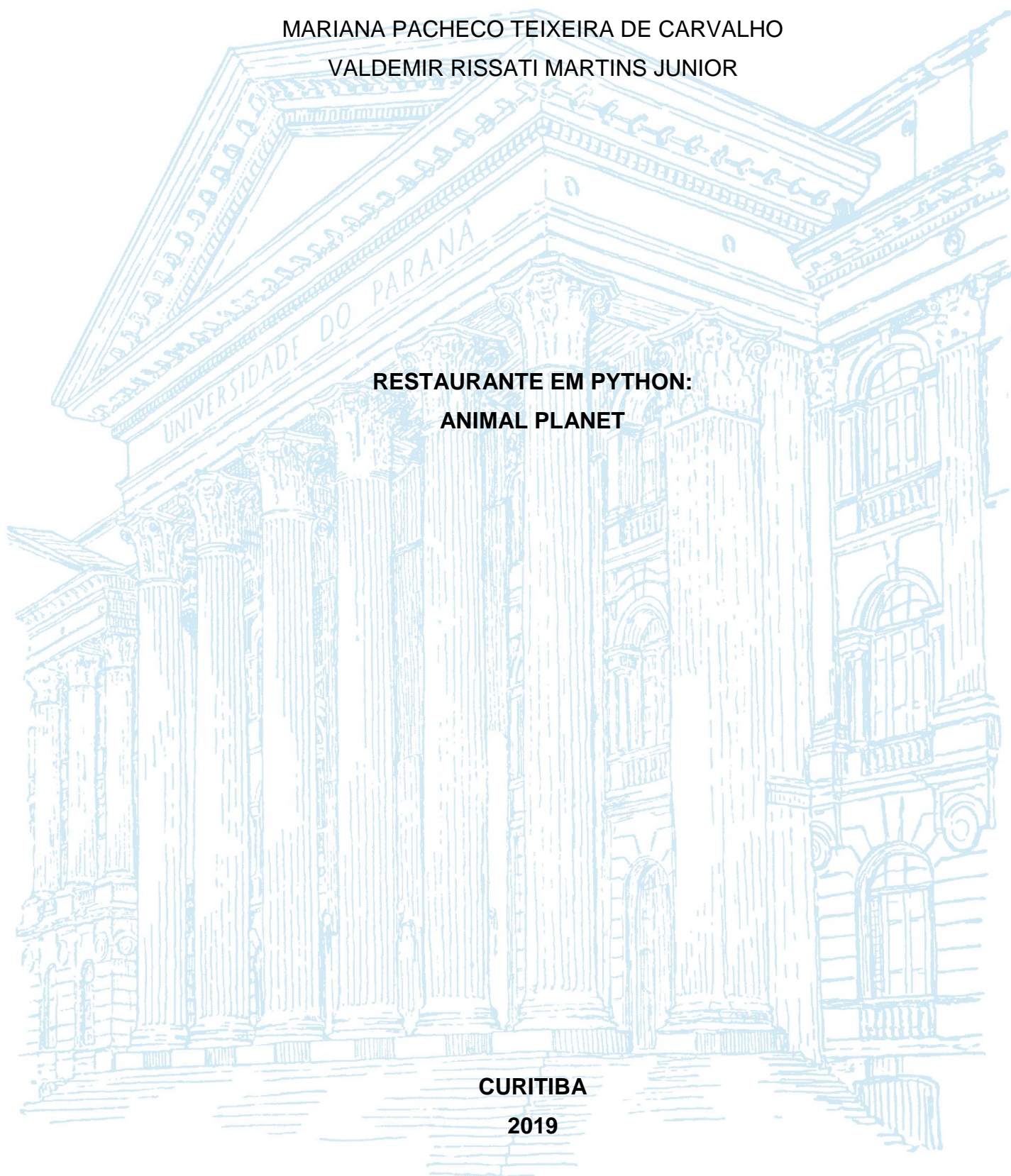
MARIANA PACHECO TEIXEIRA DE CARVALHO

VALDEMIR RISSATI MARTINS JUNIOR

**RESTAURANTE EM PYTHON:
ANIMAL PLANET**

CURITIBA

2019



VICTOR LUIZ RODRIGUES SILVA (GRR20194170)
MARIANA PACHECO TEIXEIRA DE CARVALHO (GRR20196422)
VALDEMIR RISSATI MARTINS JUNIOR (GRR20194196)

**RESTAURANTE EM PYTHON:
ANIMAL PLANET**

Relatório apresentado à disciplina
Fundamentos de Programação de
Computadores do Curso de Graduação
em Agronomia da Universidade Federal
do Paraná.

Orientador: Prof Jackson Antônio Prado
Lima

Curitiba, junho de 2019

LISTA DE FIGURAS

| | |
|---|---|
| Figura 1 – Biblioteca “telebot” | 3 |
| Figura 2 – Comando inicial /start ou “começar” | 3 |
| Figura 3.1 – Função de boas vindas e cardápio..... | 3 |
| Figura 3.1 – Função de boas vindas e cardápio..... | 4 |
| Figura 4 – Relação dos lanches que podem ser pedidos..... | 4 |
| Figura 5 – Comando “sair” e mensagem de erro..... | 4 |

SUMÁRIO

1 INTRODUÇÃO1

2 OBJETIVOS.....2

3 DESENVOLVIMENTO3

4 CONCLUSÃO6

1 INTRODUÇÃO

O Python é uma linguagem de programação de alto nível, muito usada no Linux e em outros sistemas Unix por ser muito rápida de aprender, mesmo para quem não possui experiência anterior com programação. A sintaxe é bastante organizada e a linguagem inclui suporte a objetos. Para a presente atividade, a problematização utilizada será um sistema que gerencia um restaurante.

2 OBJETIVOS

O objetivo deste trabalho é fornecer um bot em Telegram de um restaurante. A ideia foi facilitar ao usuário realizar seu pedido, de maneira simples, rápida e prática, para que não haja maiores dificuldades na sua utilização.

3 DESENVOLVIMENTO

O desenvolvimento do trabalho será em linguagem python, como já relatado anteriormente. O intermediário será um cliente pessoa física, que realizará o pedido dentre as opções disponíveis. Para a realização do código, a biblioteca “telebot” foi necessária:

```
import telebot
```

Figura 1 – Biblioteca “telebot”

Inicialmente, é dado o comando para iniciar o bot no telegram. O comando /start é utilizado no aplicativo como botão “começar”, que serve justamente para iniciar a leitura do bot requerido.

```
@bot.message_handler(commands=['start', 'ajuda'])
```

Figura 2 – Comando inicial /start ou “começar”

Após isto, o programa utiliza uma função para executar o sistema. Trata-se da função send_welcome, que mostra uma mensagem de boas vindas ao usuário, e em seguida dá as opções do cardápio.

```
def send_welcome(message):  
    """Essa função serve para dar boas vindas ao usuário :param message: Mensagem enviada pelo usuário:  
    início ou ajuda :return: retorna uma resposta ao usuário"""  
    bot.reply_to(message,  
        "Bem vindo(a) a Hamburgueria Animal Planet! Aqui você encontrará diversos tipos de  
hamburgueres. Para finalizar o processo, digite 'Sair'")  
    bot.reply_to(message, ("""Basta escolher as opções que oferecemos em nosso cardápio:  
01-X-SALADA | PREÇO: R$ 12,99 |(PÃO, DOIS HAMBURGUER, TOMATE, CEBOLA, MOSTARDA,  
MAIONESE E MOLHO BRANCO)  
02-X-CROQUI | PREÇO: R$ 13,99 |(PÃO, DOIS HAMBURGUER, TOMATE, CEBOLA FRITA  
CROCANTE, MOSTARDA E MAIONESE)  
03-X-MÉXICO | PREÇO: R$ 14,99 |(PÃO, DOIS HAMBURGUER, DUPLO COMBO DE PIMENTA,  
CEBOLA, ALFACE E MOLHO DE PIMENTA)  
04-X-RATIOU | PREÇO: R$ 15,99 |(PÃO, DOIS HAMBURGUER, TOMATE, ISCAS DE FRANGO AO  
MOLHO BRANCO)  
05-X-PIG | PREÇO: R$ 16,99 |(PÃO, DOIS HAMBURGUER, BACON CROCANTE, TOMATE,  
ALECRIM E PIMENTA)  
06-X-CHICKEN | PREÇO: R$ 17,99 |(PÃO, DOIS HAMBURGUER, TOMATE, PIMENTA, OREGANO E  
FRANGO DESFIADO)  
07-X-VINA | PREÇO: R$ 18,99 |(PÃO, DOIS HAMBURGUER, TOMATE, DUAS VINAS E  
MAIONESE VERDE)  
08-X-KING | PREÇO: R$ 19,99 |(PÃO, DOIS HAMBURGUER, TOMATE, BACON, CEBOLA,  
MAIONESE E VINA)  
09-X-4QUEIJOS | PREÇO: R$ 20,99 |(PÃO, DOIS HAMBURGUER, TOMATE, BACON, DUPLO QUEIJO  
AO MOLHO BRANCO DERRETIRO E OREGANO)
```

Figura 3.1 – Função de boas vindas e cardápio

```

10-X-CASA      | PREÇO: R$ 21,99 |(PÃO, TRÊS HAMBURGUER, DUPLO BACON, VINA, PIMENTA,
MOLHO BRANCO E FRANGO DESFIADO)
11-SUCO DE LARANJA(500ML)      | PREÇO: R$ 6, 99
12 - SUCO DE LIMÃO(500ML)      | PREÇO: R$ 6, 99
13 - SUCO DE TAMARINDO(500ML) | PREÇO: R$ 6, 99
14 - FANTA UVA(350ML)          | PREÇO: R$ 4, 50
15 - FANTA JESUS(350ML)        | PREÇO: R$ 4, 50
16 - COCA COLA(2L)              | PREÇO: R$ 8, 00
17-MOUSSE DE LIMÃO              |PREÇO: R$ 4,00
18-TORTA DE LIMÃO               |PREÇO: R$ 3,50
19-PAMONHA                      |PREÇO: R$ 4,99
20-COCADA COM DOCE DE LEITE    |PREÇO: R$ 4,99
21-BRIGADEIRO DE POTE           |PREÇO: R$ 6,99""))

```

Figura 3.2 – Função de boas-vindas e cardápio

Após receber as opções de cardápio, o usuário pode escolher qualquer uma delas. Cada vez que escolhe uma opção, esta é contabilizada ao valor total da compra. Uma mensagem é enviada para cada lanche que é pedido.

```

@bot.message_handler(func=lambda message: True)
def echo_all(message):
    if message.text.lower() == '01':
        bot.reply_to(message, "Um X-SALADA saindo!")
    elif message.text.lower() == '02':
        bot.reply_to(message, "Um X-CROQUI saindo!")
    elif message.text.lower() == '03':
        bot.reply_to(message, "Um X-MÉXICO saindo!")
    elif message.text.lower() == '04':
        bot.reply_to(message, "Um X-RATIOU saindo!")
    elif message.text.lower() == '05':
        bot.reply_to(message, "Um X-PIG saindo!")
    elif message.text.lower() == '06':
        bot.reply_to(message, "Um X-CHICKEN saindo!")
    elif message.text.lower() == '07':
        bot.reply_to(message, "Um X-VINA saindo!")
    elif message.text.lower() == '08':
        bot.reply_to(message, "Um X-KING saindo!")
    elif message.text.lower() == '09':
        bot.reply_to(message, "Um X-4QUEIJOS saindo!")
    elif message.text.lower() == '10':
        bot.reply_to(message, "Um X-CASA saindo!")

```

Figura 4 – Relação dos lanches que podem ser pedidos

Ao todo são 21 opções, dentre elas hambúrgueres, sobremesas e bebidas. Caso o cliente queira finalizar o pedido, basta digitar “Sair”, que será direcionada uma mensagem com os detalhes do pedido. Caso seja digitada uma opção que não está entre as relacionadas, aparece a seguinte mensagem: “Opção inválida. Tente novamente...”.

```

elif message.text.lower() == 'SAIR' or message.text.lower() == 'Sair' or message.text.lower() == 'sair':
    bot.reply_to(message, "Obrigada por escolher o nosso restaurante, esperamos que voce goste, sua comida
ja esta a caminho!")
else:
    bot.reply_to(message, ('Opção invalida! Tente novamente...'))

```

Figura 5 – Comando “sair” e mensagem de erro

Finalmente, o comando `bot.polling()` finaliza a sessão.

```
bot.polling() #finaliza o programa (obrigatorio)
```

Figura 6 – Comando “bot.polling” finaliza a sessão

4 CONCLUSÃO

Programar em python ou em qualquer outra ferramenta de programação não é uma tarefa fácil. Utilizamos os conhecimentos aprendidos em sala e, como o tempo foi curto, fizemos o melhor possível.

Tivemos vários problemas com o código. Por exemplo, ao enviar o total dos pedidos e a listagem deles, aparecia TOTAL = 0 E LISTA = []. Não conseguimos somar os valores. Enfim, encontramos vários empecilhos durante a montagem do código.

Temos certeza que devemos continuar melhorando e aperfeiçoando nossos conhecimentos, além que foi aprendido neste semestre.

Segue abaixo o que deveria ser o código original:

```
import telebot
#baixa-se o telebot pelo pip do python no prompt de comando: pip
install pyTelegramBotAPI
TOKEN = '785570680:AAEEdjvCFWLABJtfYxiX4rsU_xyRYRcqpsc'
#o codigo que identifica meu bot no telegram
bot = telebot.TeleBot(TOKEN)

@bot.message_handler(commands=['start', 'ajuda'])
def send_welcome(message):
    """Essa função serve para dar boas vindas ao usuário :param
    mensagem: Mensagem enviada pelo usuário: inicio ou ajuda :return:
    retorna uma resposta ao usuário"""
    bot.reply_to(message,
        "Bem vindo(a) a Hamburgueria Animal Planet! Aqui você
    encontrará diversos tipos de hamburgueres.Para finalizar o processo,
    digite 'Sair'")
    bot.reply_to(message, ("Basta escolher as opções que oferecemos
    em nosso cardapio:
                                01-X-SALADA | PREÇO: R$ 13 |(PÃO, DOIS
HAMBURGUER, TOMATE, CEBOLA, MOSTARDA, MAIONESE E MOLHO BRANCO)
                                02-X-CROQUI | PREÇO: R$ 14 |(PÃO, DOIS
HAMBURGUER, TOMATE, CEBOLA FRITA CROCANTE, MOSTARDA E MAIONESE)
                                03-X-MÉXICO | PREÇO: R$ 15 |(PÃO, DOIS
HAMBURGUER, DUPLO COMBO DE PIMENTA, CEBOLA, ALFACE E MOLHO DE PIMENTA)
                                04-X-RATIOU | PREÇO: R$ 16 |(PÃO, DOIS
HAMBURGUER, TOMATE, ISCAS DE FRANGO AO MOLHO BRANCO)
                                05-X-PIG | PREÇO: R$ 17 |(PÃO, DOIS
HAMBURGUER, BACON CROCANTE, TOMATE, ALECRIM E PIMENTA)
                                06-X-CHICKEN | PREÇO: R$ 18 |(PÃO, DOIS
HAMBURGUER, TOMATE, PIMENTA, OREGANO E FRANGO DESFIADO)
                                07-X-VINA | PREÇO: R$ 19 |(PÃO, DOIS
HAMBURGUER, TOMATE, DUAS VINAS E MAIONESE VERDE)
                                08-X-KING | PREÇO: R$ 20 |(PÃO, DOIS
HAMBURGUER, TOMATE, BACON, CEBOLA, MAIONESE E VINA)
                                09-X-4QUEIJOS| PREÇO: R$ 21 |(PÃO, DOIS
HAMBURGUER, TOMATE, BACON, DUPLO QUEIJO AO MOLHO BRANCO DERRETIRO E
OREGANO)
                                10-X-CASA | PREÇO: R$ 22 |(PÃO, TRÊS
HAMBURGUER, DUPLO BACON, VINA, PIMENTA, MOLHO BRANCO E FRANGO
DESEFIADO)
                                11-SUCO DE LARANJA(500ML) | PREÇO: R$
7
                                12 - SUCO DE LIMÃO(500ML) | PREÇO: R$
7
                                13 - SUCO DE TAMARINDO(500ML) | PREÇO: R$
7
                                14 - FANTA UVA(350ML) | PREÇO: R$
5
```

```

7
7
7
5
5
8, 00
4,00

12 - SUCO DE LIMÃO(500ML) | PREÇO: R$
13 - SUCO DE TAMARINDO(500ML) | PREÇO: R$
14 - FANTA UVA(350ML) | PREÇO: R$
15 - FANTA JESUS(350ML) | PREÇO: R$
16 - COCA COLA(2L) | PREÇO: R$
17-MOUSSE DE LIMÃO |PREÇO: R$
18-TORTA DE LIMÃO |PREÇO: R$ 4
19-PAMONHA |PREÇO: R$ 5
20-COCADA COM DOCE DE LEITE |PREÇO: R$ 5
21-BRIGADEIRO DE POTE |PREÇO: R$

7"")) #cardapio

@bot.message_handler(func=lambda message: True)

```

```

def echo_all(message): #se o cliente digitar o numero correspondente,
o restaurante armazena o pedido .
    TOTAL = 0 #com isso, o codigo funciona, mas o problema é que ele
ira zerar toda vez que passar para o proximo if elif else,
    # entao no final o preço total = 0 e a lista vai ficar vazia, nao
mostrando o que o cliente pediu.(desafio do trabalho e complicações)
    lista = []
    if message.text.lower() == '01':
        TOTAL+=13 #soma dos precos caso o cliente escolha essa opção
        lista.append('X-SALADA') #armazenamento do pedido em uma lista
para informa-lo no final
        bot.reply_to(message, "Um X-SALADA saindo!")
    elif message.text.lower() == '02':
        TOTAL+=14
        lista.append('X-CROQUI')
        bot.reply_to(message, "Um X-CROQUI saindo!")
    elif message.text.lower() == '03':
        TOTAL+=15
        lista.append('X-MÉXICO')
        bot.reply_to(message, "Um X-MÉXICO saindo!")
    elif message.text.lower() == '04':
        TOTAL+=16
        lista.append('X-RATIOU')
        bot.reply_to(message, "Um X-RATIOU saindo!")
    elif message.text.lower() == '05':
        TOTAL+=17
        lista.append('X-PIG')
        bot.reply_to(message, "Um X-PIG saindo!")
    elif message.text.lower() == '06':
        TOTAL+=18
        lista.append('X-CHICKEN')
        bot.reply_to(message, "Um X-CHICKEN saindo!")
    elif message.text.lower() == '07':
        TOTAL+=19
        lista.append('X-VINA')
        bot.reply_to(message, "Um X-VINA saindo!")
    elif message.text.lower() == '08':
        TOTAL+=20
        lista.append('X-KING')
        bot.reply_to(message, "Um X-KING saindo!")
    elif message.text.lower() == '09':
        TOTAL+=21
        lista.append('X-4QUEIJOS')
        bot.reply_to(message, "Um X-4QUEIJOS saindo!")
    elif message.text.lower() == '10':
        TOTAL+=22
        lista.append('X-CASA')
        bot.reply_to(message, "Um X-CASA saindo!")
    elif message.text.lower() == '11':
        TOTAL+=7

```

```

        lista.append('SUCO DE LARANJA')
        bot.reply_to(message, "Seu SUCO DE LARANJA foi anotada!")
    elif message.text.lower() == '12':
        TOTAL+=7
        lista.append('SUCO DE LIMÃO')
        bot.reply_to(message, "Seu SUCO DE LIMÃO foi anotada")
    elif message.text.lower() == '13':
        TOTAL+=7
        lista.append('SUCO DE TAMARINDO')
        bot.reply_to(message, "Seu SUCO DE TAMARINDO foi anotada")
    elif message.text.lower() == '14':
        TOTAL+=5
        lista.append('FANTA UVA')
        bot.reply_to(message, "Sua FANTA UVA foi anotada")
    elif message.text.lower() == '15':
        TOTAL+=5
        lista.append('FANTA JESUS')
        bot.reply_to(message, "Sua FANTA JESUS foi anotada")
    elif message.text.lower() == '16':
        TOTAL+=8
        lista.append('COCA COLA')
        bot.reply_to(message, "Sua COCA COLA foi anotada")
    elif message.text.lower() == '17':
        TOTAL+=4
        lista.append('MOUSSE DE LIMÃO')
        bot.reply_to(message, "Ótima escolha! Um MOUSSE DE LIMÃO para adoçar seu dia! ")
    elif message.text.lower() == '18':
        TOTAL+=4
        lista.append('TORTA DE LIMÃO')
        bot.reply_to(message, "Ótima escolha! Um TORTA DE LIMÃO para adoçar seu dia! ")
    elif message.text.lower() == '19':
        TOTAL+=5
        lista.append('PAMONHA')
        bot.reply_to(message, "Ótima escolha! Um PAMONHA para adoçar seu dia! ")
    elif message.text.lower() == '20':
        TOTAL+=5
        lista.append('COCADA COM DOCE DE LEITE')
        bot.reply_to(message, "Ótima escolha! Um COCADA COM DOCE DE LEITE para adoçar seu dia! ")
    elif message.text.lower() == '21':
        TOTAL+=7
        lista.append('BRIGADEIRO DE POTE')
        bot.reply_to(message, "Ótima escolha! Um BRIGADEIRO DE POTE para adoçar seu dia! ")
    elif message.text.lower() == 'Finalizar' or message.text.lower() == 'FINALIZAR' or message.text.lower() == 'finalizar': #garantir que qualquer 'finaliza' o programa aceite
        bot.reply_to(message, "Obrigada por escolher o nosso restaurante, esperamos que voce goste, sua comida já esta a caminho! O preço final ficou:{0}".format(TOTAL)) #mostraria o total
        bot.reply_to(message,"Voce pediu:{0}".format(lista))
        #mostraria o que o cliente pediu no total
    else:
        bot.reply_to(message, ('Opção invalida! Tente novamente...'))
        #caso o cliente digite algo fora do comum.

bot.polling() #finaliza o programa (obrigatorio)

```