

INSTITUTO SUPERIOR POLITÉCNICO INTERCONTINENTAL DE LUANDA

DEPARTAMENTO DE ENGENHARIA

CURSO DE INFORMÁTICA

2º Ano/Turno: Manhã/Turma: EIM2

**AGENDA TELEFÓNICA**

**Por:**

Grupo D

**DOCENTE**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**ANTUNES VITI**

**Ano académico/2023-2024**



INSTITUTO SUPERIOR POLITÉCNICO INTERCONTINENTAL DE LUANDA

DEPARTAMENTO DE ENGENHARIA

CURSO DE INFORMÁTICA

2º Ano/Turno: Manhã/Turma: EIM2

**AGENDA TELEFÓNICA**

Trabalho de carácter avaliativo a ser entregue ao docente da cadeira de Linguagem de Programação.

Docente:

Lic. **Antunes Viti**

**Ano académico/2023-2024**

**INTEGRANTES**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **INTEGRANTES** | | | | |
| **Nº** | **NOME COMPLETO** | **E-mail** | **NOTA DO TRABALHO** | **NOTA DA DEFESA** |
| **01** | Aguinaldo Lencastre | [clevandrolencastre@gmail.com](mailto:clevandrolencastre@gmail.com) |  |  |
| **02** | António Alberto | [antonioalberto14@mail.com](mailto:antonioalberto14@mail.com) |  |  |
| **03** | João Augusto | [johnnguifa6@gmail.com](mailto:johnnguifa6@gmail.com) |  |  |
| **04** | Sulamita Fitiboa | [sulamitafitiboa@gmail.com](mailto:sulamitafitiboa@gmail.com) |  |  |
| **05** | Simão Guimarães | [simondahollywood@gmail.com](mailto:simondahollywood@gmail.com) |  |  |
| **06** | William Kahumba | [willianngonga@gmail.com](mailto:willianngonga@gmail.com) |  |  |
| **07** | Edgar Luta | [edgarluta63@gmail.com](mailto:edgarluta63@gmail.com) |  |  |

**INTRODUÇÃO**

O presente trabalho tem como finalidade debruçar sobre Agenda telefônica. Sendo que lista telefônica ou Agenda telefônica é uma publicação destinada à divulgação de informações sobre assinantes do serviço de [telefonia](https://pt.wikipedia.org/wiki/Telefonia). Basicamente uma lista telefônica é composta de três seções principais: lista de assinantes (por ordem de nomes), lista classificada (por produtos e atividades) e lista de endereços (por logradouros).

**PARTE PRÁTICA DO PROJECTO**

Inventado o telefone em 1876 e disseminado o seu uso, logo se fez sentir a necessidade por parte de quem o possuía da relação de quem também o possuísse.

Passando rapidamente na explicação do projeto. Vale expressar que o nosso projeto, é constituído por três (3) classes associadas entre si, que poderão compor a Agenda telefônica como já foi referido anteriormente.

A primeira classe é denominada Endereço, tendo como atributos: rua, casa, município, província e país.

class Endereco:  
 def \_\_init\_\_(self, rua, casa, municipio, provincia, pais):  
 self.rua = rua  
 self.casa = casa  
 self.municipio = municipio  
 self.provincia = provincia  
 self.pais = pais

A segunda classe é denominada Contato, tendo como atributos: nome, telefone, e-mail, endereço, e bi.

class Contato:  
 def \_\_init\_\_(self, nome, telefone, email, bi, endereco):  
 self.nome = nome  
 self.telefone = telefone  
 self.email = email  
 self.endereco = endereco  
 self.bi = bi

E a terceira e última é a classe Agenda, nessa classe, esta classe que pegará os atributos das classes anteriores, sendo parte dela, pois um contacto ou um endereço não tem a sua agenda, porém, a agenda contem contactos e endereços. Daí a relação das classes anteriores.

Não obstante destacarmos as classes que compõem a agenda, a mesma dispõem-se de um menu inicial, que por, e através dele são iniciados a cada uma das opções da agenda. Tais como:

1. Adicionar contato;
2. Editar contato;
3. Remover contato;
4. Buscar contato por nome;
5. Buscar contato por telefone;
6. Sair

Mais do que falar, precisamos demostrar à prática. Assim sendo, deixamos abaixo o código como anexo, composto por 119 linhas de códigos. Passando para a apresentação do seu funcionamento.

|  |
| --- |
| print("==============================================================================") print(" AGENDA TELEFÓNICA ") print("============================================================================\n") class Endereco:  def \_\_init\_\_(self, rua, casa, municipio, provincia, pais):  self.rua = rua  self.casa = casa  self.municipio = municipio  self.provincia = provincia  self.pais = pais  class Contato:  def \_\_init\_\_(self, nome, telefone, email, bi, endereco):  self.nome = nome  self.telefone = telefone  self.email = email  self.bi = bi  self.endereco = endereco  class Agenda:  def \_\_init\_\_(self):  self.contatos = []   def adicionar\_contato(self, contato):  self.contatos.append(contato)   def editar\_contato(self, nome, novo\_contato):  for i in range(len(self.contatos)):  if self.contatos[i].nome == nome:  self.contatos[i] = novo\_contato  break   def remover\_contato(self, nome):  for contato in self.contatos:  if contato.nome == nome:  self.contatos.remove(contato)  break   def buscar\_contato\_por\_nome(self, nome):  for contato in self.contatos:  if contato.nome == nome:  return contato  return None   def buscar\_contato\_por\_telefone(self, telefone):  for contato in self.contatos:  if contato.telefone == telefone:  return contato  return None  def menu\_inicial():  print("Apresentamos-lhe a agenda telefonica | Escolha as opções abaixos para começar a usar\n")  print("1. Adicionar contato")  print("2. Editar contato")  print("3. Remover contato")  print("4. Buscar contato por nome")  print("5. Buscar contato por telefone")  print("6. Sair")  opcao = int(input("Escolha uma opção: "))  return opcao  # Exemplo de uso do menu agenda = Agenda() while True:  opcao = menu\_inicial()  if opcao == 1:  # Adicionar contato  nome = input("Nome do contato: ")  telefone = input("Telefone do contato: ")  email = input("Email do contato: ")  bi=input("BI ")  rua = input("Rua: ")  casa = input("Casa: ")  municipio = input("Município: ")  provincia = input("Província: ")  pais = input("País: ")  endereco = Endereco(rua, casa, municipio, provincia, pais)  novo\_contato = Contato(nome, telefone, email, bi, endereco)  agenda.adicionar\_contato(novo\_contato)  elif opcao == 2:  # Editar contato  nome = input("Nome do contato a ser editado: ")  novo\_nome = input("Novo nome do contato: ")  novo\_telefone = input("Novo telefone do contato: ")  novo\_email = input("Novo email do contato: ")  novo\_rua = input("Nova rua: ")  novo\_casa = input("Nova casa: ")  novo\_bi=input("Novo BI ")  novo\_municipio = input("Novo município: ")  novo\_provincia = input("Nova província: ")  novo\_pais = input("Novo país: ")  novo\_endereco = Endereco(novo\_rua, novo\_casa, novo\_municipio, novo\_provincia, novo\_pais)  novo\_contato = Contato(novo\_nome, novo\_telefone, novo\_email, novo\_endereco)  agenda.editar\_contato(nome, novo\_contato)  elif opcao == 3:  # Remover contato  nome = input("Nome do contato a ser removido: ")  agenda.remover\_contato(nome)  elif opcao == 4:  # Buscar contato com o nome  nome = input("Nome do contato a ser buscado: ")  contato\_encontrado = agenda.buscar\_contato\_por\_nome(nome)  if contato\_encontrado:  print(f"Contato encontrado: {contato\_encontrado.nome} - {contato\_encontrado.telefone}")  else:  print("Contato não encontrado")  elif opcao == 5:  # Buscar contato com nª de telefone  telefone = input("Telefone do contato a ser buscado: ")  contato\_encontrado = agenda.buscar\_contato\_por\_telefone(telefone)  if contato\_encontrado:  print(f"Contato encontrado: {contato\_encontrado.nome} - {contato\_encontrado.telefone}")  else:  print("Contato não encontrado")  elif opcao == 6:  # para finalizar o programa  break  else:  print |

**CONCLUSÃO**

Durante a elaboração do trabalho, pelas pesquisas desenvolvidas, convenientemente concluímos que uma agenda telefônica bem preenchida com todos os elementos necessários, permite a fácil detenção e localização de indivíduos listado à nossa Agenda de contactos.