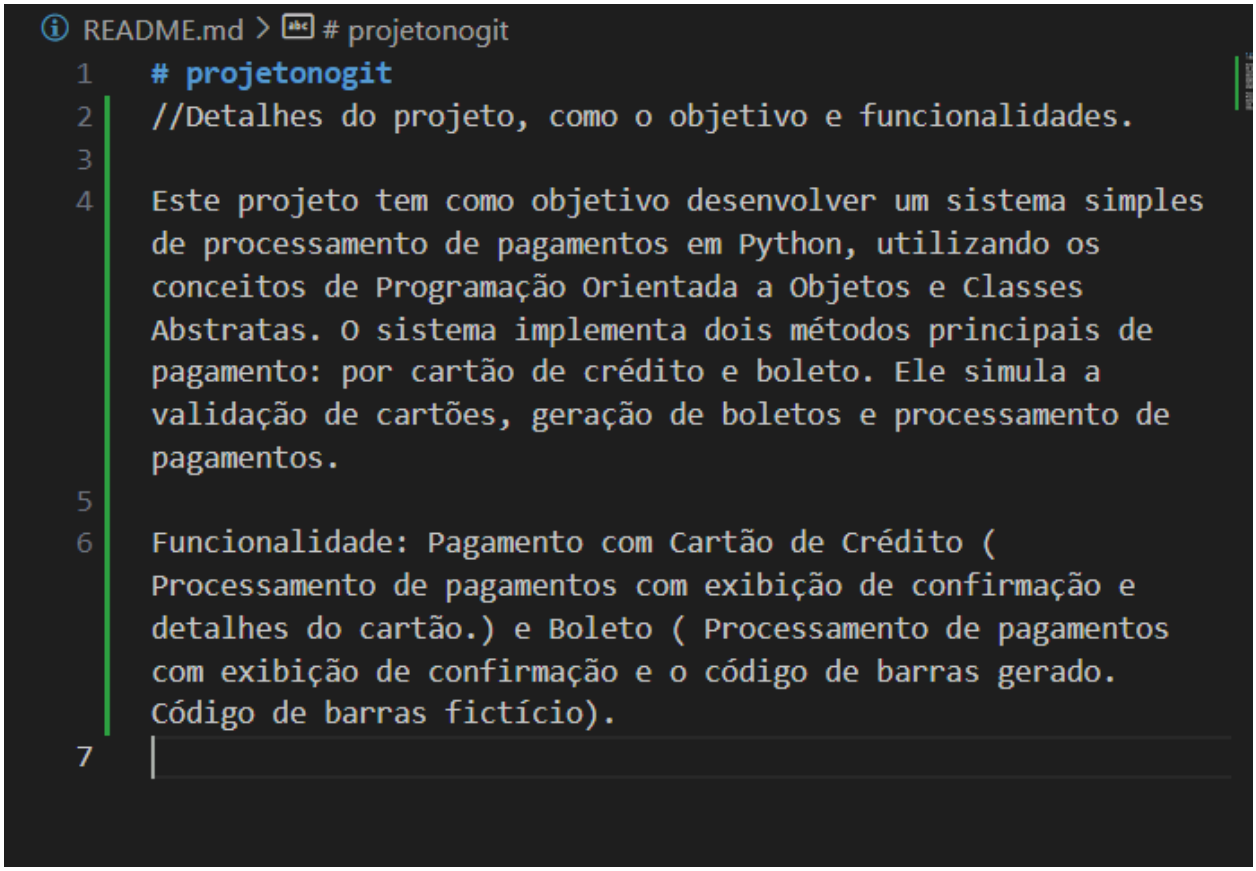
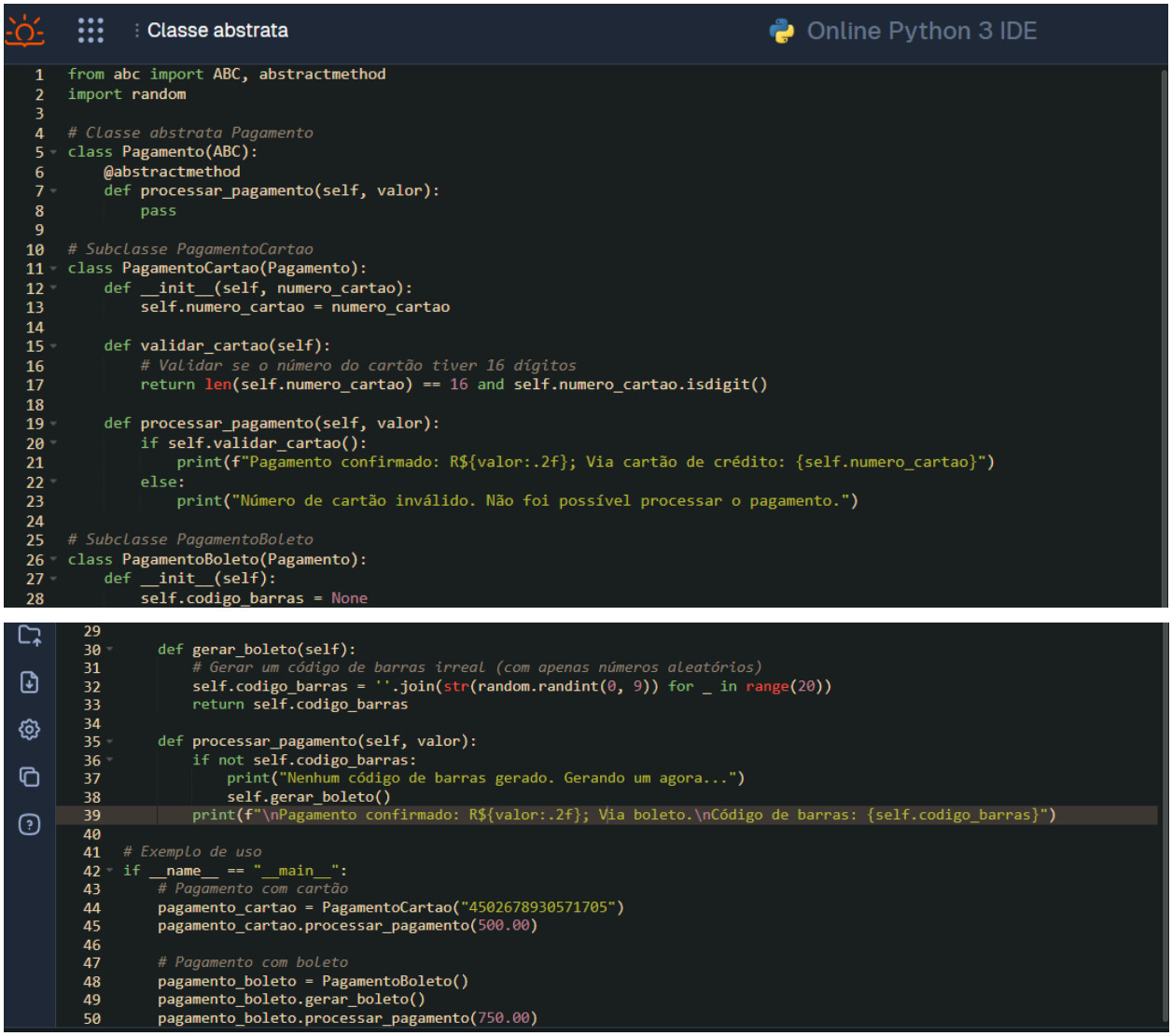
DOCUMENTAÇÃO



No processo de desenvolvimento, primeiramente foi preciso criar um arquivo chamado **“README.md”,** onde se encontram as informações do projeto, como o seu objetivo principal seria a criação de sistema de dados na linguagem **Python**. Conforme a POO, utilizará conceitos de classes **Abstratas.** O sistema se baseia em duas formas de pagamento, o cartão de crédito e por boleto.

E sua funcionalidade seria as formas de pagamento, onde seriam exibidas as informações do cartão (caso seja pago no crédito), e os detalhes do boleto, como a geração de código de barras fictício.

formapagamento.py




```python
from abc import ABC, abstractmethod
import random

# Classe abstrata Pagamento
class Pagamento(ABC):
    @abstractmethod
    def processar_pagamento(self, valor):
        pass

# Subclasse PagamentoCartao
class PagamentoCartao(Pagamento):
    def __init__(self, numero_cartao):
        self.numero_cartao = numero_cartao

    def validar_cartao(self):
        # Validar se o número do cartão tiver 16 dígitos
        return len(self.numero_cartao) == 16 and self.numero_cartao.isdigit()

    def processar_pagamento(self, valor):
        if self.validar_cartao():
            print(f"Pagamento confirmado: R${valor:.2f}; Via cartão de crédito: {self.numero_cartao}")
        else:
            print("Número de cartão inválido. Não foi possível processar o pagamento.")

# Subclasse PagamentoBoleto
class PagamentoBoleto(Pagamento):
    def __init__(self):
        self.codigo_barras = None

    def gerar_boleto(self):
        # Gerar um código de barras irreal (com apenas números aleatórios)
        self.codigo_barras = ''.join(str(random.randint(0, 9)) for _ in range(20))
        return self.codigo_barras

    def processar_pagamento(self, valor):
        if not self.codigo_barras:
            print("Nenhum código de barras gerado. Gerando um agora...")
            self.gerar_boleto()
        print(f"\nPagamento confirmado: R${valor:.2f}; Via boleto.\nCódigo de barras: {self.codigo_barras}")

# Exemplo de uso
if __name__ == "__main__":
    # Pagamento com cartão
    pagamento_cartao = PagamentoCartao("4502678930571705")
    pagamento_cartao.processar_pagamento(500.00)

    # Pagamento com boleto
    pagamento_boleto = PagamentoBoleto()
    pagamento_boleto.gerar_boleto()
    pagamento_boleto.processar_pagamento(750.00)
```

Logo em seguida, o arquivo onde se encontra o sistema (**formapagamento.py**). A classe abstrata é a classe **Pagamento.** E as subclasses **PagamentoCartao** e **PagamentoBoleto,** que herdam os métodos da classe Abstrata. E o sistema possui funções para processar o pagamento e validar o cartão.

DESAFIOS

Criar um projeto no Git, foi sem dúvidas um desafio, pois exige concentração e pelo fato que alguns erros acabam aparecendo. Mas o Git foi uma forma de ajuda para separar por etapas todo o projeto, tornando o gerenciamento mais eficiente.