

Classes abstratas

Python tem suporte à criação de classes abstratas do paradigma orientado à objetos, porém sua sintaxe é um pouco diferente de linguagens como Java, C# ou C++.

Qual das opções abaixo cria uma classe abstrata com pelo menos um método abstrato?

Selecione uma alternativa

A

```
# -*- coding: UTF-8 -*-
from abc import ABCMeta, abstractmethod

class Boleto(object):
    __metaclass__ = ABCMeta

    def imprime(self):
        linha_digitável = self.gera_linha_digitavel()
        # usa a linha digitável para construir e imprimir o recibo

    @abstractmethod
    def gera_linha_digitavel(self):
        pass
```

B

```
# -*- coding: UTF-8 -*-
from abc import ABCMeta, abstractmethod

class Boleto(object):

    def imprime(self):
        linha_digitável = self.gera_linha_digitavel()
        # usa a linha digitável para construir e imprimir o recibo

    @abstractmethod
    def gera_linha_digitavel(self):
        pass
```

C

```
# -*- coding: UTF-8 -*-
from abc import ABCMeta, abstractmethod

class Boleto(object):
    __metaclass__ = ABCMeta

    def imprime(self):
        linha_digitável = self.gera_linha_digitavel()
        # usa a linha digitável para construir e imprimir o recibo

    def gera_linha_digitavel(self):
        pass
```

D

```
# -*- coding: UTF-8 -*-
from abc import ABCMeta

class Boleto(object):
    __metaclass__ = ABCMeta

    def imprime(self):
        linha_digitável = self.gera_linha_digitavel()
        # usa a linha digitável para construir e imprimir o recibo

    @abstractmethod
    def gera_linha_digitavel(self):
        pass
```