

AD1_PIG_2021_2

Generated by Doxygen 1.9.2

1 Namespace Index	1
1.1 Namespace List	1
2 Class Index	3
2.1 Class List	3
3 File Index	5
3.1 File List	5
4 Namespace Documentation	7
4.1 AD1_2021_2 Namespace Reference	7
4.1.1 Function Documentation	7
4.1.1.1 main()	7
4.2 desconto_racional Namespace Reference	8
4.3 met_numerico Namespace Reference	8
4.3.1 Function Documentation	8
4.3.1.1 df()	8
4.3.1.2 dfe()	8
4.3.1.3 f()	9
4.3.1.4 fe()	9
4.3.1.5 metodo_newton()	9
5 Class Documentation	11
5.1 desconto_racional.Financiamento Class Reference	11
5.1.1 Detailed Description	11
5.1.2 Constructor & Destructor Documentation	11
5.1.2.1 __init__()	12
5.1.3 Member Function Documentation	12
5.1.3.1 __str__()	12
5.1.3.2 fator_k()	12
5.1.3.3 numero_parcelas()	12
5.1.3.4 percentual_a_mais()	13
5.1.3.5 recomendacao_pagamento()	13
5.1.3.6 taxa_real()	13
5.1.3.7 valor_financiado()	13
5.1.3.8 valor_presente()	13
5.1.3.9 valor_prestacao()	14
5.1.4 Member Data Documentation	14
5.1.4.1 CF	14
5.1.4.2 e	14
5.1.4.3 p	14
5.1.4.4 t	14
5.1.4.5 x	14
5.1.4.6 y	15

5.2 desconto_racional.Tabela_Price Class Reference	15
5.2.1 Detailed Description	15
5.2.2 Constructor & Destructor Documentation	15
5.2.2.1 __init__()	15
5.2.3 Member Function Documentation	16
5.2.3.1 criar_tabela()	16
5.2.3.2 imprimir_tabela()	16
5.2.4 Member Data Documentation	16
5.2.4.1 data	16
5.2.4.2 np	16
5.2.4.3 pmt	16
5.2.4.4 pv	16
5.2.4.5 t	16
6 File Documentation	17
6.1 AD1_2021_2.py File Reference	17
6.2 desconto_racional.py File Reference	17
6.3 met_numerico.py File Reference	17
Index	19

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

AD1_2021_2	7
desconto_racional	8
met_numerico	8

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

desconto_racional.Financiamento	11
desconto_racional.Tabela_Price	15

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

AD1_2021_2.py	17
desconto_racional.py	17
met_numerico.py	17

Chapter 4

Namespace Documentation

4.1 AD1_2021_2 Namespace Reference

Functions

- def `main` (`argv=None`)

4.1.1 Function Documentation

4.1.1.1 `main()`

```
def AD1_2021_2.main (
    argv = None )
```

Desconto Racional por Dentro

Os módulos `argparse` e `tabulate` foram utilizadas na implementação deste código.
Para instalar as dependências, use:

```
- pip install -r requirements.txt
```

:argument: argumentos da linha de comando:

```
-h (--help) help
-p (--parcelas) número de parcelas.
-t (--taxa) taxa mensal (Taxa SELIC por padrão).
-x (--valorP) valor da compra à prazo.
-y (--valorV) valor da compra à vista.
-e (--entrada) indica se houve entrada (False por padrão).
```

Uso:

```
- AD1_2021_2.py -p 10 -t 1 -x 500 -y 450 -e True
- AD1_2021_2.py -p 18 -t 0 -x 3297.60 -y 1999
- AD1_2021_2.py -p 10 -x 1190 -y 1094.80 -e True
- AD1_2021_2.py -p 8 -t 4.55 -x 111064.80 -y 23000
- AD1_2021_2.py -p 9 -t 0 -x 134788.8 -y 63816.24
- AD1_2021_2.py -p 4 -t 3.0 -x 1080.11 -y 1000
- AD1_2021_2.py --parcelas 14 --taxa 4.55 --valorP 14000.50 --valorV 23000 --entrada True
- AD1_2021_2.py --help
```

4.2 desconto_racional Namespace Reference

Classes

- class [Financiamento](#)
- class [Tabela_Price](#)

4.3 met_numerico Namespace Reference

Functions

- def [metodo_newton](#) (x, y, p, ti, i, e)
- def [fe](#) (x, y, p, t)
- def [dfe](#) (x, y, p, t)
- def [f](#) (x, y, p, t)
- def [df](#) (x, y, p, t)

4.3.1 Function Documentation

4.3.1.1 df()

```
def met_numerico.df (
    x,
    y,
    p,
    t )
```

Derivada da função auxiliar para o método de Newton sem entrada.

4.3.1.2 dfe()

```
def met_numerico.dfe (
    x,
    y,
    p,
    t )
```

Derivada da função auxiliar para o método de Newton com entrada.

4.3.1.3 f()

```
def met_numerico.f (
    x,
    y,
    p,
    t )
```

Função auxiliar para o método de Newton sem entrada.

4.3.1.4 fe()

```
def met_numerico.fe (
    x,
    y,
    p,
    t )
```

Função auxiliar para o método de Newton com entrada.

4.3.1.5 metodo_newton()

```
def met_numerico.metodo_newton (
    x,
    y,
    p,
    ti,
    i,
    e )
```

Método de Newton para encontrar a taxa real.

```
:param x: valor à prazo
:param y: valor à vista
:param p: número de parcelas
:param ti: taxa inicial
:param i: número de iterações
:param e: entrada

:return: taxa final e número de iterações realizadas
```


Chapter 5

Class Documentation

5.1 desconto_racional.Financiamento Class Reference

Public Member Functions

- def [__init__](#) (self, [p](#), [x](#), [y](#), [t](#), [e](#))
- def [__str__](#) (self)
- def [valor_prestacao](#) (self)
- def [valor_presente](#) (self)
- def [fator_k](#) (self)
- def [valor_financiado](#) (self)
- def [numero_parcelas](#) (self)
- def [taxa_real](#) (self)
- def [recomendacao_pagamento](#) (self)
- def [percentual_a_mais](#) (self)

Public Attributes

- [p](#)
- [x](#)
- [y](#)
- [t](#)
- [e](#)
- [CF](#)

5.1.1 Detailed Description

Classe `Financiamento` contém todas as informações referentes ao financiamento realizado.

5.1.2 Constructor & Destructor Documentation

5.1.2.1 `__init__()`

```
def desconto_racional.Financiamento.__init__ (
    self,
    p,
    x,
    y,
    t,
    e )
```

Construtor da classe financiamento para inicializar o objeto.

```
:param p: numero de parcelas
:param x: valor à prazo
:param y: valor à vista
:param t: taxa mensal
:param e: entrada
```

5.1.3 Member Function Documentation

5.1.3.1 `__str__()`

```
def desconto_racional.Financiamento.__str__ (
    self )
```

Representa o objeto do tipo financiamento como string.

5.1.3.2 `fator_k()`

```
def desconto_racional.Financiamento.fator_k (
    self )
```

Calcula o fator que produz o preço atualizado no instante da compra.

```
:return: fator k
```

5.1.3.3 `numero_parcelas()`

```
def desconto_racional.Financiamento.numero_parcelas (
    self )
```

Calcula o número de parcelas restantes do financiamento.

```
:return: número de parcelas
```


5.1.3.4 percentual_a_mais()

```
def desconto_racional.Financiamento.percentual_a_mais (
    self )
```

Retorna o percentual pago a mais no caso do valor total corrigido ser maior que o valor a vista.

5.1.3.5 recomendacao_pagamento()

```
def desconto_racional.Financiamento.recomendacao_pagamento (
    self )
```

Imprime uma recomendação de pagamento com base na comparação entre a taxa mensal fornecida no momento do financiamento e a taxa real aplicada.

5.1.3.6 taxa_real()

```
def desconto_racional.Financiamento.taxa_real (
    self )
```

Acha a taxa que produz o preço à vista pelo método de Newton.

:return: taxa real e número de iterações

5.1.3.7 valor_financiado()

```
def desconto_racional.Financiamento.valor_financiado (
    self )
```

Calcula o valor financiado, o que depende se houve entrada ou não.

:return: valor financiado

5.1.3.8 valor_presente()

```
def desconto_racional.Financiamento.valor_presente (
    self )
```

Calcula o preço atualizado no instante da compra.

:return: valor presente

5.1.3.9 valor_prestacao()

```
def desconto_racional.Financiamento.valor_prestacao (
    self )
```

Calcula o valor das prestações do financiamento.

:return: valor de cada prestação

5.1.4 Member Data Documentation

5.1.4.1 CF

```
desconto_racional.Financiamento.CF
```

5.1.4.2 e

```
desconto_racional.Financiamento.e
```

5.1.4.3 p

```
desconto_racional.Financiamento.p
```

5.1.4.4 t

```
desconto_racional.Financiamento.t
```

5.1.4.5 x

```
desconto_racional.Financiamento.x
```

5.1.4.6 y

`desconto_racional.Financiamento.y`

The documentation for this class was generated from the following file:

- [desconto_racional.py](#)

5.2 desconto_racional.Tabela_Price Class Reference

Public Member Functions

- `def __init__ (self, np, pv, t, pmt)`
- `def criar_tabela (self)`
- `def imprimir_tabela (self)`

Public Attributes

- `np`
- `pv`
- `t`
- `pmt`
- `data`

5.2.1 Detailed Description

Classe Tabela_Price contém todas as operações necessárias para imprimir a tabela.

5.2.2 Constructor & Destructor Documentation

5.2.2.1 __init__()

```
def desconto_racional.Tabela_Price.__init__ (
    self,
    np,
    pv,
    t,
    pmt )
```

Construtor da classe TabelaPrice para inicializar o objeto.

```
:param np: numero de parcelas
:param pv: valor financiado
:param t: taxa mensal
:param pmt: valor de cada parcela
```

5.2.3 Member Function Documentation

5.2.3.1 criar_tabela()

```
def desconto_racional.Tabela_Price.criar_tabela (
    self )

Realiza os cálculos necessários para criar a tabela price

:return: matriz com os dados da tabela price
```

5.2.3.2 imprimir_tabela()

```
def desconto_racional.Tabela_Price.imprimir_tabela (
    self )

Imprime a tabela price utilizando uma matriz de dados.
```

5.2.4 Member Data Documentation

5.2.4.1 data

```
desconto_racional.Tabela_Price.data
```

5.2.4.2 np

```
desconto_racional.Tabela_Price.np
```

5.2.4.3 pmt

```
desconto_racional.Tabela_Price.pmt
```

5.2.4.4 pv

```
desconto_racional.Tabela_Price.pv
```

5.2.4.5 t

```
desconto_racional.Tabela_Price.t
```

The documentation for this class was generated from the following file:

- [desconto_racional.py](#)

Chapter 6

File Documentation

6.1 AD1_2021_2.py File Reference

Namespaces

- namespace [AD1_2021_2](#)

Functions

- def [AD1_2021_2.main](#) (argv=None)

6.2 desconto_racional.py File Reference

Classes

- class [desconto_racional.Financiamento](#)
- class [desconto_racional.Tabela_Price](#)

Namespaces

- namespace [desconto_racional](#)

6.3 met_numerico.py File Reference

Namespaces

- namespace [met_numerico](#)

Functions

- def [met_numerico.metodo_newton](#) (x, y, p, ti, i, e)
- def [met_numerico.fe](#) (x, y, p, t)
- def [met_numerico.dfe](#) (x, y, p, t)
- def [met_numerico.f](#) (x, y, p, t)
- def [met_numerico.df](#) (x, y, p, t)

Index

- `__init__`
 - `desconto_racional.Financiamento`, 11
 - `desconto_racional.Tabela_Price`, 15
 - `__str__`
 - `desconto_racional.Financiamento`, 12
- `AD1_2021_2`, 7
 - `main`, 7
- `AD1_2021_2.py`, 17
- `CF`
 - `desconto_racional.Financiamento`, 14
- `criar_tabela`
 - `desconto_racional.Tabela_Price`, 16
- `data`
 - `desconto_racional.Tabela_Price`, 16
- `desconto_racional`, 8
- `desconto_racional.Financiamento`, 11
 - `__init__`, 11
 - `__str__`, 12
 - `CF`, 14
 - `e`, 14
 - `fator_k`, 12
 - `numero_parcelas`, 12
 - `p`, 14
 - `percentual_a_mais`, 12
 - `recomendacao_pagamento`, 13
 - `t`, 14
 - `taxa_real`, 13
 - `valor_financiado`, 13
 - `valor_presente`, 13
 - `valor_prestacao`, 13
 - `x`, 14
 - `y`, 14
- `desconto_racional.py`, 17
- `desconto_racional.Tabela_Price`, 15
 - `__init__`, 15
 - `criar_tabela`, 16
 - `data`, 16
 - `imprimir_tabela`, 16
 - `np`, 16
 - `pmt`, 16
 - `pvt`, 16
 - `t`, 16
- `df`
 - `met_numerico`, 8
- `dfe`
 - `met_numerico`, 8
- `e`
 - `desconto_racional.Financiamento`, 14
- `f`
 - `met_numerico`, 8
- `fator_k`
 - `desconto_racional.Financiamento`, 12
- `fe`
 - `met_numerico`, 9
- `imprimir_tabela`
 - `desconto_racional.Tabela_Price`, 16
- `main`
 - `AD1_2021_2`, 7
- `met_numerico`, 8
 - `df`, 8
 - `dfe`, 8
 - `f`, 8
 - `fe`, 9
 - `metodo_newton`, 9
- `met_numerico.py`, 17
- `metodo_newton`
 - `met_numerico`, 9
- `np`
 - `desconto_racional.Tabela_Price`, 16
- `numero_parcelas`
 - `desconto_racional.Financiamento`, 12
- `p`
 - `desconto_racional.Financiamento`, 14
- `percentual_a_mais`
 - `desconto_racional.Financiamento`, 12
- `pmt`
 - `desconto_racional.Tabela_Price`, 16
- `pvt`
 - `desconto_racional.Tabela_Price`, 16
- `recomendacao_pagamento`
 - `desconto_racional.Financiamento`, 13
- `t`
 - `desconto_racional.Financiamento`, 14
 - `desconto_racional.Tabela_Price`, 16
- `taxa_real`
 - `desconto_racional.Financiamento`, 13
- `valor_financiado`
 - `desconto_racional.Financiamento`, 13
- `valor_presente`
 - `desconto_racional.Financiamento`, 13

valor_prestacao

desconto_racional.Financiamento, [13](#)

x

desconto_racional.Financiamento, [14](#)

y

desconto_racional.Financiamento, [14](#)