

dnc



Data Science & Machine Learning

# **DESAFIOS - SLOT 1**

**EXERCÍCIO**



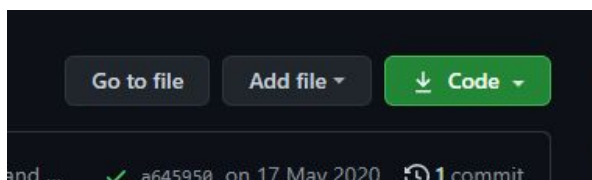
# DESAFIOS

## Bora programar!

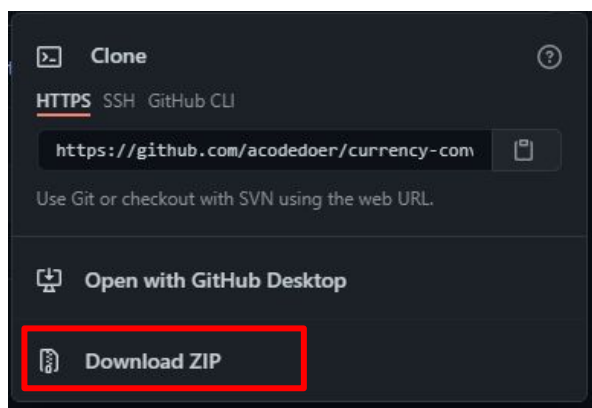
Preparamos alguns desafios para colocarem tudo que aprenderam em prática. São 3 desafios, sendo os **2 primeiros obrigatórios** e o terceiro consiste em um **desafio extra**, no qual você poderá praticar seus conhecimentos de programação e ampliar seu portfólio.

Estes desafios deverão ser entregues em uma pasta **compactada no formulário** de submissão encontrado na plataforma. Para fazer isso você pode desenvolver seu projeto em um repositório no GitHub e baixar o arquivo zipado, como mostram as imagens a seguir:

1. Clique no botão “Code”:



2. Selecione o arquivo Download ZIP



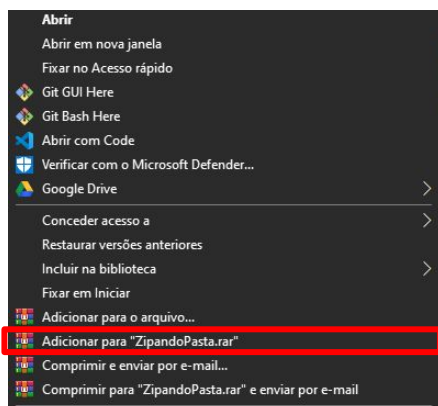


# DESAFIOS

---

## Bora programar!

Caso seus arquivos já estejam em uma pasta no seu computador você também possui a possibilidade de salvá-la como **.rar** e submeter os arquivos. Para isso basta clicar com o botão direito na pasta (que contém os arquivos) e selecionar a opção “**Adicionar para ‘NomeDoArquivo.rar’**”.



É necessário que, minimamente, os arquivos dos **desafios obrigatórios** estejam nesta pasta compactada.

Bom trabalho!

Observação: os hiperlinks contidos neste arquivo estão destacados da seguinte forma: **Exemplo**, para ajudá-lo(a) a saber onde clicar.

**DESAFIO 1**

**CALCULADORA**



# CALCULADORA

---

## Comece com o básico

Calculadoras são ferramentas extremamente úteis, sendo essenciais para diversas profissões atualmente. Usadas por milhares de pessoas, em diferentes contextos, precisam possuir uma interface amigável, ou seja, precisa ser intuitiva, simples e trazer o resultado correto de forma visual.

Programar sua calculadora, além de exercitar suas habilidades com Python, é uma ótima maneira de entender a interface do usuário e o processamento de eventos em um aplicativo. Neste desafio, você criará uma calculadora que suporta cálculos aritméticos básicos (soma, subtração, multiplicação e divisão) em números inteiros.

Comece entendendo o funcionamento e definindo as lógicas necessárias. Faça o programa rodar de forma simples, certificando-se de que:

- Qualquer pessoa seja capaz de utilizar seu programa;
- O programa não possui brechas, como a inserção de letras na operação;



# CALCULADORA

---

## Pense na interface!

Sua calculadora funciona? Com as funcionalidades básicas já implementadas, é hora de incrementar. Coloque outras funções recorrente em uma calculadora. Pode ser interessante experimentar o aplicativo 'calculadora' em seu celular para entender melhor as funcionalidades básicas e como funciona a interface. O estilo é com você, então use sua imaginação e seja criativo!

Deixamos um [LINK](#) que conta um pouco sobre a história e as funcionalidades de uma calculadora básica, caso queira tirar alguma dúvida.

Seguem algumas funcionalidades esperadas na calculadora:

- Assim que uma operação for realizada, o usuário possa inserir outra;
- O último resultado deve ficar salvo na memória, caso o usuário queira utilizá-lo para uma operação.
- O usuário deve ver uma tela mostrando o número atual inserido ou o resultado da última operação.
- O usuário deve ver um teclado de entrada contendo botões de 0 até 9, e operadores '+', '-', '/', e '=', um botão 'C' (para limpar) e um botão 'AC' (para limpar tudo).
- O usuário pode inserir números como sequências de até 8 dígitos clicando nos dígitos no teclado de entrada. A entrada de qualquer dígito superior a 8 será ignorada.
- O usuário pode clicar em um botão de operação para exibir o resultado dessa operação.
- O usuário pode clicar no botão 'C' para limpar o último número ou a última operação.
- O usuário pode clicar no botão 'AC' para limpar todas as áreas de trabalho internas e definir a exibição para 0.
- O usuário pode ver 'ERRO' exibido se qualquer operação exceder o máximo de 8 dígitos.



# CALCULADORA

---

## Algumas dicas

1. Uma biblioteca que pode ser necessária é a **PyQt5**. Leia sobre ela, algumas funções poderão ser úteis na criação da interface.
2. Talvez seja necessário importar para seu código alguns módulos, como **sys** e **re**.
3. Pesquise em repositórios no GitHub! Veja como outras pessoas fazem, pegue inspirações e crie o seu código, da sua maneira!
4. Lembre-se, se simplesmente copiar um código, você não estará aprendendo a programar... Então se esforce para criar o seu programa!

**DESAFIO 2**

**GERADOR**

**DE SENHAS**





# GERADOR DE SENHAS

---

## Garanta a proteção dos seus dados!

Segurança digital tem se tornado um tópico extremamente relevante nos dias atuais. Vivemos em um mundo completamente vinculado à tecnologia, no qual, senhas são necessárias para diversas tarefas do cotidiano, desde acessar o computador a conferir o saldo bancário.

Utilizar nomes, datas ou palavras aleatórias no dicionário não são a forma mais segura de garantir a proteção dos seus dados. Estas são senhas simples, fáceis de serem descobertas.

Um gerador de senhas aleatórias é uma ótima forma para garantir uma proteção dos seus dados! Neste desafio você irá desenvolver um gerador de senhas baseado em características selecionadas pelo usuário.

Inicie com o básico. Determine o funcionamento do seu sistema, quais serão as regras para a criação da senha. Você pode usar um sistema de substituição, no qual, a partir de uma chave inserida pelo usuário, o sistema gera uma senha substituindo a partir de uma regra definida pelo programador. Exemplo:

Se: Aa = 19; Ee = 3; Tt = %.

Chave = TesteSenha

Senha Gerada = %3s%3S3nh19

Existem outras metodologias de criptografia, busque sobre elas e defina qual irá seguir para o seu gerador de senhas.



# GERADOR DE SENHAS

---

## Personalize seu sistema.

Nem todas as plataformas aceitam o mesmo tipo de senha, por isso, você precisa garantir que seu gerador poderá ser útil para qualquer um que queira usá-lo. Com as estruturas do gerador estabelecidas, foque em garantir uma personalização do sistema, siga as seguintes exigências:

- O usuário deverá poder determinar o comprimento da senha gerada.
- O usuário pode selecionar se deseja que a senha:
  - Inclua letras maiúsculas;
  - Inclua letras minúsculas;
  - Inclua números;
  - Inclua símbolos;
- Ao clicar no botão Gerar senha, o usuário pode ver uma senha sendo gerada.
- O usuário pode clicar no botão Copiar para a área de transferência que salvará a senha na área de transferência.

Use o sistema a seguir como **benchmark**: **Gerador de Senhas**.



# GERADOR DE SENHAS

---

## Algumas dicas

1. Pesquise sobre geradores de senha e sistemas de criptografia. Uma boa dica é começar por **zxcvbn**.
2. Foque em criar o sistema de geração de senhas personalizável, sendo que, uma interface bem feita é um overdelivery (uma entrega maior que o exigido).
3. Pesquise em repositórios no GitHub! Veja como outras pessoas fazem, pegue inspirações e crie o seu código, da sua maneira!
4. Lembre-se, se simplesmente copiar um código, você não estará aprendendo a programar... Então se esforce para criar o seu programa!

**DESAFIO EXTRA**

**CONVERSOR**

**DE MOEDAS**



# CONVERSOR DE MOEDAS

---

## Quanto está o Dólar hoje?

Existem dezenas de moedas ao redor do mundo com seus valores variando diariamente, a depender da dinâmica da economia. O que define o valor da moeda em relação a outras é a Taxa de Câmbio.

Neste desafio você deve criar um sistema de conversão de moedas, no qual você poderá conferir quanto determinada moeda vale em relação a outras. Os valores das moedas devem variar conforme a cotação diária, logo, os valores não podem ser estáticos e devem ser atualizados automaticamente.

Seguem as funcionalidades esperadas:

- O usuário deve ser capaz de inserir até 9 dígitos para representar a quantidade a ser convertida em um campo;
- O usuário deve conseguir visualizar uma lista classificada de moedas disponíveis e selecionar conversão desejada;
- O usuário deve ser capaz de visualizar o valor (arredondado para duas casas decimais) da quantia de origem convertida na moeda de destino;
- O usuário deve ser alertado se a entrada não for um número.

Clique para observar um **benchmark** de aplicação: **CONVERSOR**.



# CONVERSOR DE MOEDAS

---

## Algumas dicas

1. Pesquise sobre APIs, será necessário utilizar uma funcionalidade como está para automatizar a atualização do sistema. Uma boa dica é começar por **Currency Layer**.
2. Talvez você necessite utilizar alguma biblioteca para converter os arquivos json, por exemplo, a **urllib** é ótima nesta situação.
3. Foque em criar o sistema de cotação funcional, sendo que, uma interface bem feita é um overdelivery (uma entrega maior que o exigido).
4. Pesquise em repositórios no GitHub! Veja como outras pessoas fazem, pegue inspirações e crie o seu código, da sua maneira!
5. Lembre-se, se simplesmente copiar um código, você não estará aprendendo a programar... Então se esforce para criar o seu programa!

dnc



Data Science & Machine Learning

**#HARDWORK**

**Esperamos que o material  
tenha sido útil para você!  
Foco nos estudos!**