



# Virtu-uau!

Módulo 2 - Aula 01 - HARD





## Objetivos de aprendizagem

- Compreender o que são emuladores;
- Reconhecer a importância da utilização dos ambientes virtuais, como meio de evitar que um código não rode em outra máquina;
- Definir o que são bibliotecas e saber como instalá-las;
- Construir um ambiente virtual.



Todos os direitos reservados  
©2023 Resílio Educação



**Nesta aula, veremos:**

- Emuladores;
- Ambientes virtuais;
- Bibliotecas.

**Por que aprender isso?**

- Emuladores permitem a criação de ambientes virtuais que imitam as condições reais em que os dados são coletados ou processados.
- Ambientes virtuais são uma parte essencial do desenvolvimento de software moderno e, particularmente, úteis quando vários projetos diferentes estão sendo desenvolvidos ao mesmo tempo, ou quando diferentes versões de bibliotecas ou dependências precisam ser testadas.
- As bibliotecas, por sua vez, ajudam a trabalhar com grandes conjuntos de dados, fornecendo uma ampla gama de ferramentas e funções especializadas que permitem a manipulação, transformação e visualização dos dados.



Todos os direitos reservados  
©2023 Resilio Educação

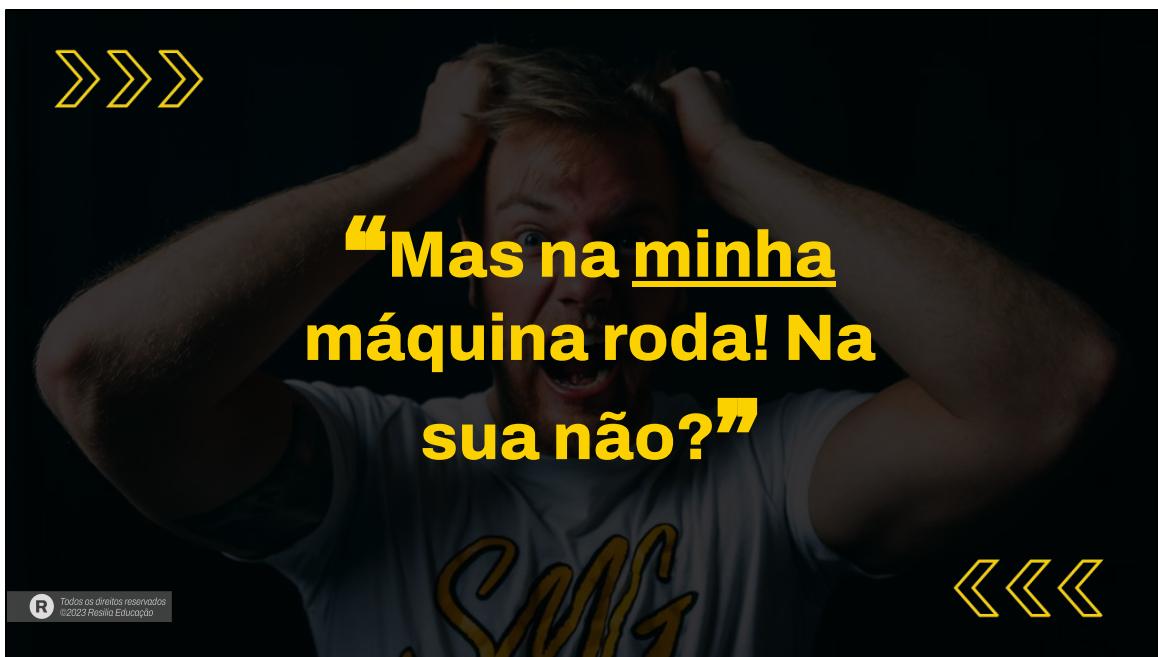


### NOTA PARA O ESTUDANTE

**Emuladores** são softwares que permitem que um sistema operacional ou dispositivo "emule" o comportamento de outro sistema ou dispositivo, de forma que o software originalmente destinado a ser executado em um sistema possa ser executado em um ambiente diferente. Em outras palavras, um emulador permite que um computador ou dispositivo execute programas ou jogos que foram projetados para serem executados em outro tipo de computador ou dispositivo.

#### **Mas como isso é feito?**

Através da replicação do hardware e software do dispositivo original, de modo que o emulador possa imitar o comportamento e funcionamento do dispositivo original.



 Todos os direitos reservados  
©2023 Resilia Educação

### NOTA PARA O ESTUDANTE

#### **Por que os programas rodam de forma diferentes em máquinas/ ambientes diferentes?**

Ambiente de desenvolvimento é onde os programadores criam e testam o código, geralmente em uma máquina local. Já o ambiente de produção é onde o software é executado para os usuários finais. A principal diferença entre eles é que o ambiente de desenvolvimento geralmente é mais flexível e permite mudanças frequentes e experimentação, enquanto o ambiente de produção precisa ser estável e confiável, evitando interrupções no funcionamento do software para os usuários.

Um código pode funcionar apenas na máquina do programador devido a diferenças em sistemas operacionais, versões de software ou dependências ausentes. Isso pode ser resolvido testando o código em um ambiente de produção que replica o ambiente do usuário final ou implementando soluções de virtualização ou contêineres para garantir que o software seja executado de maneira consistente em diferentes ambientes.



## Bibliotecas

São pacotes/módulos que contém código fonte que foi pré-compilado e empacotado como um arquivo com operações que englobam um objetivo-alvo.

As bibliotecas são uma coleção de algoritmos com a intenção de compartilhar soluções por meio de funções ou métodos que podem ser importadas assim como os módulos.

Algumas bibliotecas precisam ser instaladas e para isso deve usar o comando abaixo:

```
pip install nome_da_biblioteca
```



Todos os direitos reservados  
©2023 Resilia Educação

## GLOSSÁRIO

Uma biblioteca de software é um conjunto de algoritmos, funções e rotinas pré-escritas que podem ser reutilizadas em diferentes projetos de software. Aqui estão alguns exemplos de bibliotecas populares:

- NumPy: uma biblioteca para computação científica em Python, com suporte para matrizes e vetores de alta dimensão.
- jQuery: uma biblioteca de JavaScript que simplifica a manipulação de documentos HTML e interações com o navegador.
- TensorFlow: uma biblioteca de software livre de aprendizado de máquina desenvolvida pelo Google Brain Team, com suporte para construção e treinamento de modelos de redes neurais.
- React: uma biblioteca de JavaScript para construção de interfaces de usuário interativas e responsivas.
- OpenCV: uma biblioteca de visão computacional de código aberto, com suporte para detecção de objetos, reconhecimento de faces e muito mais.

 **RELEMBRANDO...**

Um **algoritmo** é um conjunto de instruções passo a passo que descrevem como realizar uma tarefa ou resolver um problema. É um processo sistemático para realizar uma operação ou solucionar um problema, geralmente expresso em uma sequência lógica de etapas ou passos.

# Ambientes virtuais



R Todos os direitos reservados  
©2020 Resello Educação

## Ambientes virtuais

Um **ambiente virtual** é um **local** preparado especialmente para um determinado projeto onde o interpretador, as bibliotecas e os scripts instalados nele ficam **isolados** do resto do sistema do computador.

Ali só ficarão instaladas as bibliotecas que colocamos daqui pra frente.

Os ambientes virtuais são popularmente conhecidos como **environments** em Python.

 Todos os direitos reservados  
©2023 Resilia Educação

### NOTA PARA O ESTUDANTE

Em Python, environments (ambientes) são espaços isolados onde as dependências e pacotes de um projeto podem ser instalados e gerenciados separadamente de outros projetos e do ambiente global do sistema. Isso é útil para desenvolvedores que trabalham em vários projetos com diferentes dependências, evitando conflitos entre elas.

Os environments em Python são criados usando ferramentas como **virtualenv**, **conda** ou **venv**. Eles permitem que os desenvolvedores definam e gerenciem conjuntos específicos de pacotes, versões de pacotes e configurações de ambiente para cada projeto separadamente.

Quando um environment é ativado, as dependências e pacotes instalados nele são usados para o projeto em que o ambiente está sendo usado. Isso permite que os desenvolvedores mantenham versões diferentes de pacotes e dependências para projetos diferentes e possam facilmente alternar entre eles conforme necessário.

### Mas por que Python é tão importante?

O uso de ambientes virtuais em **Python** é uma prática recomendada para o

desenvolvimento de software, pois ajuda a garantir a compatibilidade e estabilidade do projeto e evita conflitos entre diferentes projetos.



## **Venv**

Em Python, utilizamos **venv** como escolha para a criação de **ambientes virtuais**. Seu uso é possível a partir da **versão 3.6** da linguagem.

Para **criar um novo ambiente**, em um terminal, você deve digitar:

```
python -m venv nome_env
```

Para **ativar** o ambiente virtual:

```
.\nome_env\Scripts\activate
```

Para **desativar**:

```
deactivate
```



Todos os direitos reservados  
©2023 Resilia Educação

## **GLOSSÁRIO**

O **venv** é uma ferramenta que permite criar um ambiente virtual para desenvolver programas em Python. Isso significa que, ao invés de instalar as bibliotecas e pacotes diretamente no seu computador, você pode criar um espaço isolado para cada projeto em que estiver trabalhando. Isso é útil porque cada projeto pode precisar de versões diferentes de bibliotecas e pacotes, e manter tudo separado ajuda a evitar conflitos.

Ao criar um ambiente virtual com o **venv**, você pode instalar as bibliotecas e pacotes necessários para o seu projeto sem interferir em outras instalações de Python em seu computador. Além disso, você pode facilmente compartilhar o ambiente virtual com outras pessoas para garantir que todos estejam trabalhando com as mesmas versões de bibliotecas e pacotes.



# Requirements



 Todos os direitos reservados  
© 2023 Rosalia Educapção

# **Requirements**

Os requirements (ou requisitos) são arquivos de texto designados a projetos indicando quais bibliotecas e suas versões estão sendo utilizadas.

Serve como uma “documentação”, com o intuito de guardar uma lista de pacotes/itens a serem instalados para rodar uma determinada aplicação.

Para instalar os requisitos, usamos o comando:

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



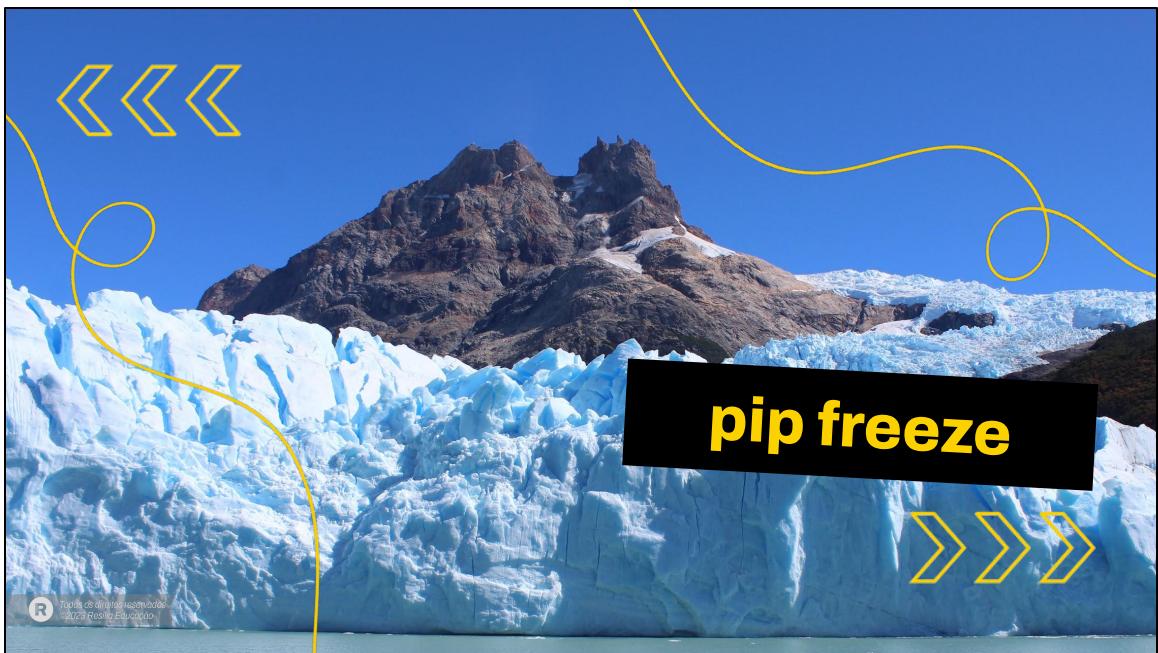
## **EXEMPLIFICANDO..**

Projeto: Gerenciador de Tarefas

Requisitos:

1. O sistema deve permitir a criação, edição e exclusão de tarefas.
2. Cada tarefa deve ter um título, descrição e data de conclusão.
3. O sistema deve permitir que o usuário filtre as tarefas por data de conclusão.
4. O sistema deve permitir que o usuário marque uma tarefa como concluída.
5. O sistema deve permitir que o usuário visualize uma lista de todas as tarefas pendentes.
6. O sistema deve permitir que o usuário visualize uma lista de todas as tarefas concluídas.
7. O sistema deve armazenar as informações das tarefas em um banco de dados.
8. O sistema deve usar uma biblioteca de gerenciamento de banco de dados, como SQLAlchemy ou Django ORM, para realizar operações de leitura e gravação no banco de dados.
9. O sistema deve ser desenvolvido em uma linguagem de programação que suporte a biblioteca de gerenciamento de banco de dados escolhida e seguir as melhores práticas de codificação e organização de código.

1. O sistema deve ser testado com diferentes cenários de criação, edição e exclusão de tarefas para verificar a correta funcionalidade do gerenciamento de banco de dados.
2. O sistema deve ser documentado adequadamente, incluindo instruções de instalação e uso da biblioteca de gerenciamento de banco de dados, para facilitar a manutenção e o suporte do software.



## Pip freeze

Para **exibir no terminal** as bibliotecas/pacotes instalados no ambiente atual e suas versões usamos o comando:

```
pip freeze
```

Para **guardarmos** as versões no arquivo **requirements.txt** sem problemas com incompatibilidade usamos o comando:

```
pip freeze > requirements.txt
```



Todos os direitos reservados  
©2023 Rosetta Educação

## GLOSSÁRIO

Pip freeze é um comando utilizado na linguagem de programação Python que exibe uma lista de todas as bibliotecas Python que estão instaladas no ambiente virtual do seu projeto, juntamente com suas respectivas versões. Isso é útil porque ajuda a garantir que outras pessoas que desejam executar o seu projeto tenham as mesmas versões de bibliotecas instaladas que você usou para desenvolvê-lo, evitando problemas de compatibilidade.

Quando você executa o comando "pip freeze" em seu ambiente virtual Python, ele produz uma lista de bibliotecas instaladas no seguinte formato:

```
makefile
Copy code
nome_da_biblioteca==versão_da_biblioteca
```

Por exemplo:

```
makefile
Copy code
Flask==1.1.2 numpy==1.20.1 pandas==1.2.2
```

Essa lista pode ser salva em um arquivo de texto para **18** que possa ser compartilhada com outras pessoas que desejam executar o seu projeto. Para criar um arquivo de

saída do comando para um arquivo de texto usando o operador "maior que" (>):

Copy code

[pip freeze > requirements.txt](#)

O arquivo "requirements.txt" resultante pode ser usado por outras pessoas para instalar todas as bibliotecas necessárias para executar o seu projeto com o mesmo ambiente virtual Python que você utilizou.

## **Atividade: Organizando a casa**

### **→ O QUE FAZER?**

Criar um ambiente virtual em um diretório do sistema e:

1. Instalar as bibliotecas Pandas e Numpy (levam 20 a 30 segundos cada);
2. Criar um código Python dentro desse ambiente e tentar importá-los dentro e fora do ambiente.

### **→ COMO FAZER?**

Em grupos de 3 a 4 pessoas.

### **→ FECHAMENTO**

Compartilhar as práticas da atividade e onde sentiram mais dificuldade.



Todos os direitos reservados  
©2023 Resilia Educação



## Para ir além

- ⇒ Neste vídeo, aprenda como criar um ambiente virtual em Python sem instalar ferramentas externas. <<https://www.youtube.com/watch?v=hA2l0TgaZhM>>
- ⇒ Neste artigo, entenda como funcionam os ambientes virtuais em Python. <<https://code.tutsplus.com/pt/tutorials/understanding-virtual-environments-in-python--cms-28272>>
- ⇒ Esse artigo traz as definições de ambientes virtuais e sua relação com os pacotes. <<https://docs.python.org/pt-br/3/tutorial/venv.html>>



Todos os direitos reservados  
©2023 Resilo Educação



»»» **Virtu-uau!**

Módulo 2 - Aula 01 - SOFT





## Objetivos de aprendizagem

- Conhecer os objetivos de aprendizagem do módulo, tanto em hard quanto em soft;
- Relembrar os combinados do curso;
- Evidenciar a importância da criação de redes e da comunidade.



Todos os direitos reservados  
©2023 Resílio Educação



**Nesta aula, veremos:**

- Conhecer o módulo;
- Redes e comunidade.

---

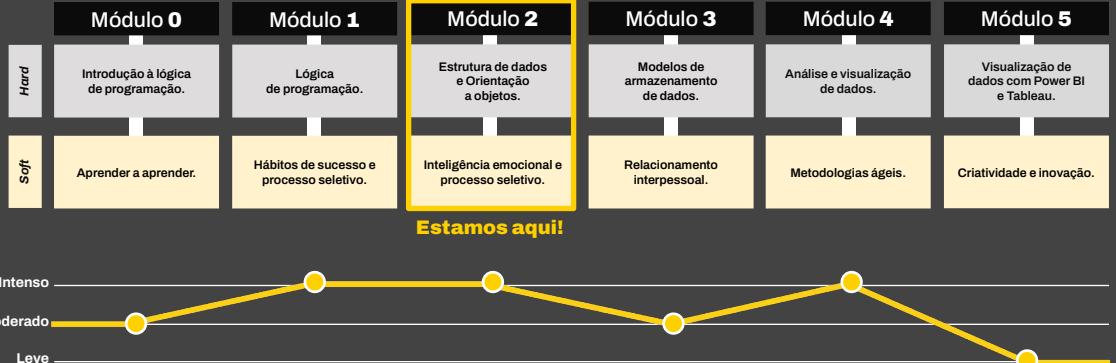
**Por que aprender isso?**

- Para você se preparar e saber de tudo que irá aprender ao longo do módulo, quais projetos irá executar e quando irá entregá-los, dessa forma você pode se planejar, criar seus ritos e rotinas e entender quando irá aprender sobre cada conteúdo que irá usar na execução dos projetos.
- As redes sociais são ideais para construir e manter relações humanas com as comunidades de marca. Grupos de discussão, fóruns ou comunidades virtuais onde as pessoas possam interagir entre si ajudam a fortalecer o relacionamento entre as pessoas em torno de um objetivo, ou/e gosto em comum.



Todos os direitos reservados  
©2023 Resílio Educação

## ONDE ESTAMOS NA JORNADA?



## MATERIAIS DE APOIO



### Por dentro da jornada

Conheça toda a estrutura do curso, a organização e os temas das aulas Soft e Hard.



### Slides

A cada aula você receberá uma cópia dos slides de aula com comentários, links e dicas a extras!



### Boas-vindas ao módulo

A cada módulo você receberá um material com um resumo do que vai encontrar no módulo.



### Escopo do projeto

Ao iniciar cada um dos projetos você terá acesso a um PDF explicativo e a um vídeo com as diretrizes do que deverá ser feito.



### E-books

Materiais complementares para leitura e divididos por temas.



### Glossário Tech

Para que você não se perca com todos os termos novos, preparamos um glossário com os principais termos do universo da tecnologia.



### Formando Squads

Entenda a dinâmica de divisão em times que você usará ao longo da jornada.



**Codewars**  
Entenda como funciona a plataforma para evoluir na habilidade de programação.

Tudo isso você vai encontrar no portal!



## Módulo 2 - Objetivos de aprendizagem *Hard Skills*



Ao final desse módulo, você deverá ser capaz de:

- ✓ Aplicar técnicas de programação para interagir com estruturas de dados, métodos e funções com Python;
- ✓ Importar e acessar as bibliotecas mais usadas no mercado;
- ✓ Utilizar ambientes virtuais para gerenciar dependências e executar melhor seus projetos.

Dica: Acesse o material “Boas-vindas ao módulo” no portal para ficar por dentro dos detalhes aula a aula.



## Módulo 2 - Objetivos de aprendizagem *Soft Skills*



**Ao final desse módulo, você deverá ser capaz de:**

- ✓ Elaborar estratégias para fazer o mapeamento de sua rede e das vagas de seu interesse;
- Compreender a estrutura e as principais etapas de um processo seletivo;
- ✓ Identificar os principais tipos de testes solicitados para desenvolvedores e reconhecer atitudes que devem e não devem ser realizadas durante a realização destes testes;
- ✓ Empregar técnicas de apresentação pessoal como meio de se destacar na etapa da entrevista em um processo seletivo;



Todos os direitos reservados  
Ministério da Educação  
www.mec.gov.br



## Módulo 2 - Objetivos de aprendizagem *Soft Skills*



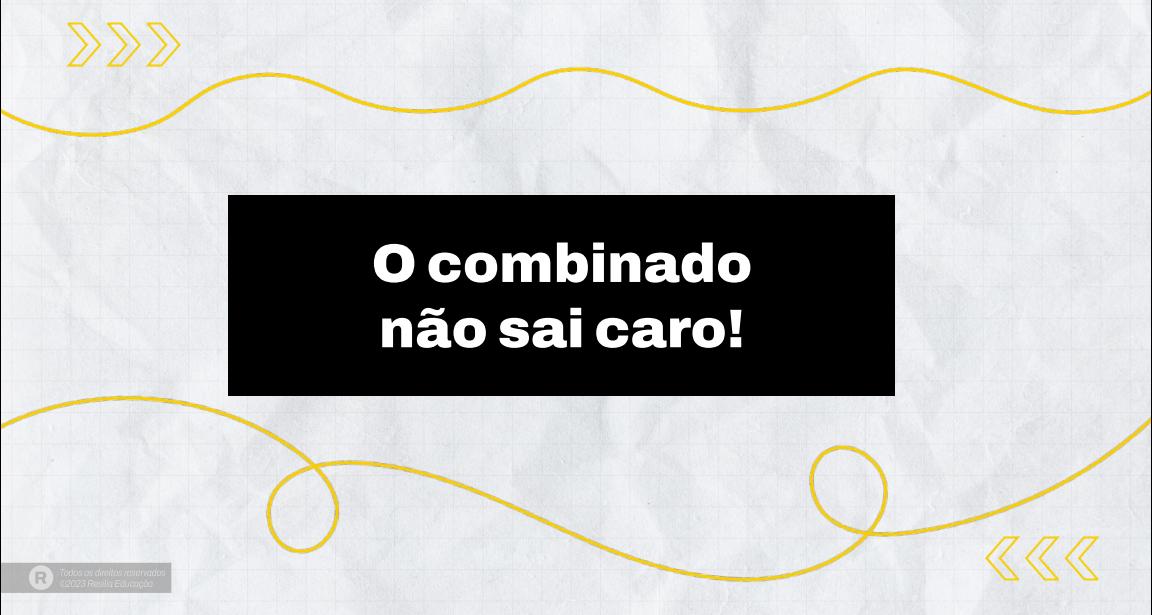
**Ao final desse módulo, você deverá ser capaz de:**

- ✓ Conhecer ferramentas para acompanhar todas as candidaturas feitas, mantendo as informações mais importantes de cada processo seletivo sempre à mão;
- ✓ Aplicar estratégias de autorregulação emocional durante um processo seletivo;
- ✓ Entender a importância de se manter motivado, principalmente quando algo não se concretiza como o esperado, em um processo seletivo;
- ✓ Reconhecer a importância da empatia em um processo seletivo, tanto para com o recrutador, quanto para com os demais candidatos.



Todos os direitos reservados  
www.cnpq.br/centro-educativo





**O combinado  
não sai caro!**



Todos os direitos reservados  
©2023 Resilia Educação



## Posturas esperadas:

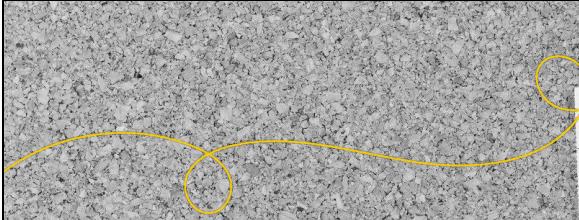
- Presença **ativa** nos encontros;
- Ser **responsável** pelo seu **desenvolvimento**;
- Pedir ajuda caso sinta dificuldade;
- Ser ético (a);
- Contar com o **apoio de seus colegas e da turma** para superar as dificuldades;
- Ser **tolerante com seus erros**, pois eles fazem parte da **aprendizagem**;
- **Tratar de forma respeitosa** seus colegas e pessoas facilitadoras em qualquer situação;
- **Buscar** conteúdos em outros sites e fóruns para **complementar sua aprendizagem**;
- **Trilhar sua jornada** tendo as **documentações** como melhores amigas;
- Comparar o seu desempenho apenas com você mesmo;
- **Praticar, praticar e praticar!**

R Todos os direitos reservados  
©2023 Resilia Educação





## Momento de reflexão



## Momento de reflexão

R 00000000000000000000000000000000

Reflitam, individualmente, o quanto cada um de vocês está **cumprindo com os combinados feitos no início do curso.**

Depois, pensem **quais são os combinados que precisam ser reforçados** nesse módulo e como vocês podem fazer isso.



**Como serão as  
avaliações do módulo?**





## Avaliações do módulo

### Projetos

- **Projeto individual** é obrigatório que deverá ser entregue na aula 8;
- **Desafio extra** e opcional, que você pode começar a fazer assim que finalizar o projeto individual;
- **Projeto em grupo (squads)** obrigatório, com entrega e apresentação na aula final do módulo.

### Lista de Atividades

- Lista de atividades divididas por temas que você encontra no Portal;
- Deve ser feita até a última aula do módulo;
- Objetivo: praticar exercícios e atividades propostas e reforçar habilidade de comprometimento e organização com as tarefas.





## Falando em projetos...

### PROJETO INDIVIDUAL

Módulo 2 - Deu match!



> Entrega: Aula 8

### DESAFIO EXTRA

Módulo 2 - Contratado!



> Opcional

### PROJETO EM GRUPO

Módulo 2 - Quero os dados na minha mesa



> Entrega: Aula 19

Dica: No portal você encontra o material um PDF + vídeo com a explicação detalhada de cada um desses projetos!





## Divisão das Squads!

Pessoa Gestora de Conhecimento

Pessoa Colaboradora I



Pessoa Co-facilitadora



Pessoa Gestora de Gente e Engajamento



Pessoa Colaboradora II

Dica: No portal você encontra o material “Formando Squads”, com a explicação detalhada e a função de cada integrante.





R Todos os direitos reservados  
©2023 Resale Educação

## **Atividade: Tamo Junto**

### **→ O QUE FAZER?**

**Responder às perguntas e desenvolver escuta ativa:**

1. Qual o maior desafio que você superou no último módulo?
2. Qual hábito você mudou ou criou desde o início do curso? Como este hábito te ajudou?
3. O que você fez durante o módulo 1 que vai levar para o módulo 2?

### **→ COMO FAZER?**

Em grupos de 4 a 5 pessoas.

### **→ FECHAMENTO**

Compartilhar as experiências.



Todos os direitos reservados  
©2023 Resilia Educação



Até a próxima e  
**#confianoprocess**