

EasyAnsi

Cores no terminal de forma simples

Para quem e este guia

- Iniciantes em Python que querem saida colorida sem configuracao complexa.
- Professores e estudantes que preferem sintaxe parecida com f-string.
- Quem precisa de mensagens de status, titulos e logging colorido.

Instalacao

```
pip install py-easyansi

# ou a partir do codigo-fonte:
pip install -e C:/caminho/para/EasyAnsi
```

Requisitos: Python 3.8+ e terminal com suporte ANSI (Windows 10+ ja funciona).

Indice do guia

Use este indice para ir direto ao tema. Cada secao traz exemplos copiaveis e diagramas para facilitar o estudo.

Por que usar EasyAnsi	3
Sintaxe //cor/texto/cor	3
activate() vs eprint/einput	4
Atalhos e mensagens de status	4
Titulos, paint e f-strings	5
Estilos combinados e cores hex	5
Logging colorido e preview()	6
Comportamento inteligente	6
Exercicio pratico + cola rapida	6

Por que usar EasyAnsi?

A EasyAnsi foi criada para quem quer cor no terminal sem aprender uma biblioteca pesada. Você escreve a cor dentro da string, como numa f-string.

EasyAnsi x bibliotecas mais complexas

Necessidade	EasyAnsi	Alternativas pesadas
Começar rápido	<code>from easyansi import eprint</code>	Configurar Console, markup
Colorir uma palavra	<code>//red/{valor}/red</code>	Montar objetos Text
Mensagem de sucesso	<code>success('Pronto')</code>	Criar renderizavel
Log colorido	<code>setup_logging()</code>	Handlers, temas, plugins
Tag escrita errada	Vira texto normal	Pode quebrar o programa

Filosofia da biblioteca

Menos conceitos, menos imports.
Comandos fáceis de lembrar no dia a dia.
Fail-safe: URLs e tags desconhecidas não quebram a saída.

Sintaxe `//cor/texto/cor`

Abertura e fechamento

Padrao	Significado	Exemplo
<code>//nome/</code>	Abre tag de estilo	<code>//red/</code>
<code>/nome</code>	Fecha tag explicitamente	<code>/red</code>
<code>//</code>	Fecha ultima tag aberta	<code>//green/texto//</code>

```
from easyansi import eprint

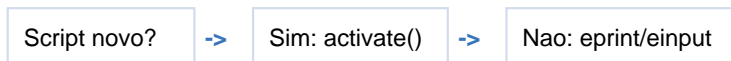
eprint('//green/Ola, mundo!/green')
eprint('Sobrou //red/1/red item no estoque')
eprint('Acesse https://exemplo.com') # URL intacta
```

Fail-safe

Somente nomes conhecidos viram cor.
`//erro/texto/erro` vira texto puro (não quebra).
URLs com `//` permanecem literais.

activate() vs eprint/einput

Fluxo:



Com `activate()`, `print` e `input` normais passam a interpretar as tags EasyAnsi:

```
import easyansi
easyansi.activate()

print('//green/Tudo certo!/green')
nome = input('Digite //cyan/seu nome/cyan: ')
easyansi.deactivate() # restaura originais
```

Alternativa explicita (sem alterar print global):

```
from easyansi import eprint, einput, fmt

eprint('//green/Tudo certo!/green')
nome = einput('Digite //cyan/seu nome/cyan: ')
mensagem = fmt('//yellow/aviso/yellow')
```

Atalhos e mensagens de status

Atalhos de cor - ideais para autocomplete no editor:

```
from easyansi import red, green, yellow, blue, bold

print(red('erro'), green('ok'), yellow('aviso'))
print(bold(red('critico'))) # encadeamento
```

Mensagens prontas para o dia a dia:

```
from easyansi import success, error, warning, info

success('Salvo com sucesso')
error('Falha ao conectar')
warning('Conexao lenta')
info('Rodando na porta 8080')
```

Referencia rapida de atalhos

Funcao	Uso tipico
red(), green(), ...	Destacar palavras isoladas
bold(), italic(), ...	Enfatizar titulos e notas
success(), error()	Feedback de operacao
style('bold-blue', txt)	Combo personalizado

Titulos, paint e f-strings

```
from easyansi import title, t, paint, ask, activate

activate()
print(title('CADASTRO DE ALUNOS', '='))
print(t('MENU', '~', style='bold-blue'))
print(paint('Texto inteiro azul', 'bold-blue'))
nome = ask('Qual seu nome?', prompt_style='bold-blue')
```

Funciona dentro de f-strings com variaveis:

```
pontos = 42
print(f'Resultado: //green/{pontos}/green pontos')
print(f'Nota //bold-blue/{7.5}/bold-blue em Matematica')
```

Dica sobre input colorido

O terminal não colore em tempo real o que você digita.
ask() colore o prompt; a resposta aparece colorida ao imprimir depois.

Estilos combinados, fundos e hex

Combine estilos com hífen: bold-blue, italic-underline-red.

```
print('//bold-blue/cabecalho/bold-blue')
print('//italic-underline-magenta/destaque/')
print('//bg-yellow/texto em fundo amarelo/bg-yellow')
print('//#ff8800/laranja exato/#ff8800')
print('//negrito-vermelho/erro em portugues/negrito-vermelho')
```

Variacoes de cor e estilo

Tipo	Tag	Exemplo
Estilo + cor	bold-blue	//bold-blue/texto/bold-blue
Fundo	bg-yellow	//bg-yellow/aviso/bg-yellow
Hex	#ff8800	//#ff8800/laranja/#ff8800
PT	negrito-vermelho	//negrito-vermelho/erro/

Logging colorido e preview()

```
import logging
from easyansi.logging import setup_logging

setup_logging(level=logging.INFO)
logging.info('Servidor iniciado')
logging.warning('Cache desatualizado')
logging.error('Falha ao conectar')
```

Cores por nível de log

Nível	Cor padrão
DEBUG	dim
INFO	cyan
WARNING	yellow
ERROR	red
CRITICAL	bold-red

Para ver todas as cores disponíveis no terminal:

```
import easyansi
easyansi.preview()
```

Comportamento inteligente da saída

Fluxo:

Condicao	Comportamento
Terminal interativo (TTY)	Cores ANSI aplicadas
Saida em pipe / arquivo	Texto limpo (sem escape)
Variavel NO_COLOR	Texto limpo
FORCE_COLOR definido	Cores forçadas

Exercicio pratico - menu de notas

```

import easyansi
easyansi.activate()

print(easyansi.title('NOTAS DO ALUNO', '='))
print('//cyan/1/cyan Listar //cyan/2/cyan Cadastrar //cyan/0/cyan Sair')
opcao = input('Escolha: ')
if opcao == '1': print('//green/Notas carregadas/green')
elif opcao == '2': print('//yellow/Cadastro em breve/yellow')
else: print('//dim/Encerrando/dim')
  
```

Cola rapida da API

fmt · eprint · einput · activate · title · t · paint · ask · preview · red/green/bold · success/error/warning/info · setup_logging