

Opa, Muri! O erro aconteceu porque o PDF gerado tentou usar caracteres especiais (como emojis 🚀) que o fpdf padrão não suporta direto, ele usa latin-1 e não Unicode completo.

Vou gerar o PDF sem emojis e caracteres especiais para evitar esse problema, beleza? Já já te passo o arquivo limpinho e fácil de usar.

Vou refazer rapidinho. Aguenta aí.

Beleza, Muri! O problema é que o fpdf que estou usando não suporta alguns caracteres Unicode como setas (→) e outros acentos. Ele aceita só Latin-1.

Vou trocar esses caracteres por versões mais simples e remover tudo que pode dar problema. Depois, testo de novo e te passo o PDF.

Já volto!

Aqui está o conteúdo revisado com símbolos simples para evitar erros de codificação no PDF:

Comandos Docker Essenciais para Provas e Desenvolvimento

1. `docker --version`
Verificar versão do Docker
2. `docker info`
Informações gerais do Docker e seu estado
3. `docker build -t nome .`
Criar imagem Docker a partir do Dockerfile
4. `docker images`
Listar imagens locais
5. `docker rmi nome`
Remover imagem Docker
6. `docker run -d -p host:cont container`
Criar e rodar container com portas mapeadas
7. `docker ps`
Listar containers em execução
8. `docker ps -a`
Listar todos containers, incluindo parados
9. `docker stop container`
Parar container em execução

10. `docker start container`
Iniciar container parado
11. `docker restart container`
Reiniciar container
12. `docker rm container`
Remover container (parado)
13. `docker rm -f container`
Forçar remoção de container (mesmo se rodando)
14. `docker logs container`
Mostrar logs do container
15. `docker exec -it container bash`
Entrar no shell do container em execução
16. `docker-compose up -d`
Subir todos containers de um projeto Compose em background
17. `docker-compose down`
Parar e remover containers, redes e volumes do Compose
18. `docker volume ls`
Listar volumes Docker
19. `docker volume rm volume`
Remover volume Docker
20. `docker network ls`
Listar redes Docker
21. `docker network rm network`
Remover rede Docker
22. `docker inspect container`
Mostrar info detalhada do container
23. `docker system prune`
Limpar containers, imagens, volumes e redes não usados
24. `docker system prune --volumes`
Limpar tudo incluindo volumes não usados

Comandos para Resolver Erros Comuns

- Container não inicia:
`docker logs container`

- Container travado / congelado:
docker restart container
docker rm -f container
- Portas em uso / conflito:
Verificar com docker ps e parar container que usa porta
- Volume não atualiza mudanças:
Verificar bind mounts no docker run -v / volumes no compose
- Imagem muito grande:
Usar .dockerignore para ignorar arquivos desnecessários
- Container sai imediatamente:
Ver logs para entender e corrigir configuração/start script
- Docker não libera espaço:
docker system prune

Dicas para Prova

- Nomeie containers com --name para facilitar controle
- Use docker ps -a para ver containers parados antes de remover
- Para rodar em background, não esqueça o -d no docker run
- Para acessar shell do container, use docker exec -it nome bash
- docker-compose é obrigatório para multi-container
- Use docker logs para debugging
- Conheça volumes e como mapear (-v ou volumes no compose)
- Entenda o ciclo: Build, Run, Exec, Stop, Remove

Agora vou gerar o PDF sem símbolos especiais. Só um momento!

```
docker run -d -p 4000:4000 --name meu-backend meu-backend-image
```

```
docker run -d -p 27017:27017 --name mongodb -v ~/data/mongo:/data/db mongo
```

Opção	O que faz	Exemplo prático
-d	Roda o container em segundo plano (detached)	docker run -d nginx
-p host:container	Mapeia a porta da máquina host para o container	-p 3000:3000
--name	Dá um nome ao container para facilitar referência	--name meu-container
-e VAR=valor	Define variáveis de ambiente dentro do container	-e NODE_ENV=production
-v host:path:container:path	Mapeia volumes/pastas do host para dentro do container	-v \$(pwd)/data:/app/data
--rm	Remove o container automaticamente após parar	docker run --rm ubuntu bash
-it	Interativo + terminal, útil para acessar shell do container	docker run -it ubuntu bash