

INSTITUTO FEDERAL

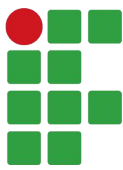
Paraná

Campus Paranaguá

Introdução

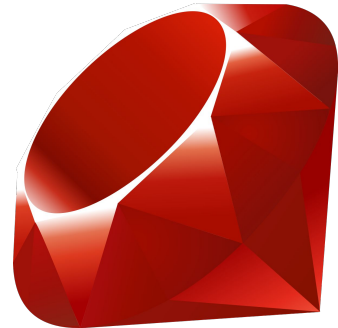
Desenvolvimento Web III

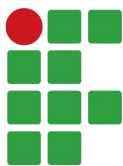
Prof. Diego Stiehl



Ruby

- Site oficial: <https://www.ruby-lang.org>
- Linguagem de Programação
 - Interpretada
 - Multiparadigma
 - Funcional, orientada a objetos, imperativa e reflexiva
 - Tipagem dinâmica e forte
 - Multiplataforma
- Criada para ser uma linguagem de script





Gems

- Gemas
 - Pedras em formato rústico
- Gems são as bibliotecas do Ruby
 - Partes de código que precisam ser “lapidadas”
 - Conjunto de arquivos reutilizáveis
- Site oficial: <https://rubygems.org>
- Em Ruby (e Rails) utilizaremos muito

Bundler

- Gerenciamento de dependências em aplicações Ruby
 - Instalação de Gems e suas dependências
- Semelhante ao Composer do PHP
- Possui um arquivo de configuração
 - Com uma linha de comando atualiza todas as dependências da aplicação



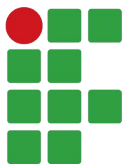
Ruby on Rails

- Framework **web** que utiliza a linguagem Ruby
 - Formado por diversas Gems (bibliotecas)
 - Projetado para tornar o desenvolvimento web mais fácil e intuitivo
- Modelo MVC bastante puro
- CoC: Convenção sobre Configuração
 - Proporciona menos código e mais recursos
- Programação orientada a diversão do programador



Ruby on Rails - Filosofias

- CoC (Convention over configuration)
 - Em vez de configurar um conjunto de arquivos, adota-se uma convenção e apenas mudar o que for necessário
- DRY (Don't Repeat Yourself)
 - Nunca fazer mais de uma vez o que for necessário (como checar uma regra de negócio)
- Automação de tarefas repetidas



Git

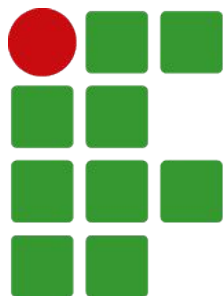
- Sistema de controle de versões
 - Open Source
 - Distribuído
- Permite trabalho de forma “desconectada”
 - Centraliza arquivos em repositórios
- Permite diversos fluxos de trabalho
- Criado por Linus Torvalds (2005)
 - Criador do Linux



GitLab

- GitHub é um host de repositórios Git na nuvem
 - <https://github.com>
- Atualmente pertence à Microsoft
- Boa parte dos projetos open source estão lá
- Caso não tenha, crie uma conta
 - Iremos usar na disciplina





INSTITUTO FEDERAL

Paraná

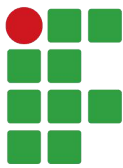
Campus Paranaguá

Configuração de Ambiente na Nuvem e Repositório Git

Mais Informações e Ferramentas

Binários e Versões

- Testar
 - `ruby -v` Versão do Ruby
 - `gem -v` Versão do RubyGems
 - `bundle -v` Versão do Bundler
 - `rails -v` Versão do Ruby on Rails
 - `git --version` Versão do Git
 - `rbenv -v` Versão do Ruby Version Manager
- Em um primeiro momento talvez não teremos tudo instalado: não es quente

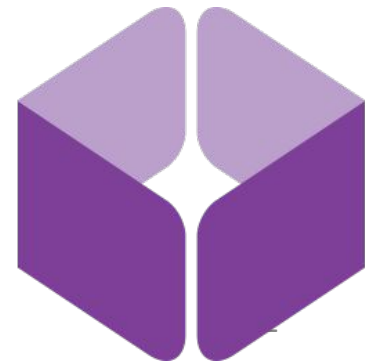


IRB

- Interactive Ruby Shell
- Permite execução direta de comandos Ruby
 - Console
- Executar:
 - `irb`
- Para sair
 - CTRL + D ou digitar “quit”

Plataforma CodeAnywhere

- Plataforma online que permite criar e utilizar máquinas virtuais
 - Gratuita para um container
- Oferece um Linux “completo”
 - Acesso de administrador
 - Conjunto de ferramentas pré-instaladas conforme necessidade
 - Como Ruby, Rails, dentre outros
- Site: <https://codeanywhere.com>



Plataforma CodeAnywhere

- Não utilizaremos o CodeAnywhere em sala
 - Mas é uma boa ferramenta caso você não possua permissão de administrador
- Ruby on Rails pronto para uso
 - Ganhamos, inclusive, uma URL pública
- Crie uma conta
 - Valide seu e-mail
 - Crie um projeto
 - Crie um container Ruby “Ruby Development Stack with RVM and Ruby on Rails preinstalled. Ubuntu 14.04”

CodeAnywhere

- Crie uma conta
 - Valide seu e-mail
- Vá para o seu [Dashboard](#)
- Crie um projeto e abra
 - Crie um container Ruby e dê um nome
 - Selecione Ruby Development Stack with RVM and Ruby on Rails preinstalled. Ubuntu 14.04

CodeAnywhere

Connection Wizard

Container

GitHub

Bitbucket

Git from URL

FTP Server

SFTP - SSH

Dropbox

Google Drive

Amazon S3

OneDrive

DigitalOcean


Name

AulaRuby

Q


ruby

All Stacks

Firehose


Firehose Ruby Development Stack with Ruby 2.1 and Apache installed.

Ubuntu 14.04

Ruby

Ruby Development Stack with RVM and Ruby on Rails preinstalled.

Ubuntu 14.04

Ruby

Ruby Development Stack with RVM and Ruby on Rails preinstalled.

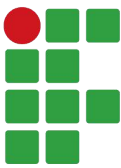
Centos 6.5

☐ Enable Always On

CREATE

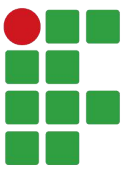
CodeAnywhere

- O container criado apresenta, dentre outros:
 - Navegador de arquivos
 - Terminal Linux (SSH Terminal)
 - Editor de texto puro com marcação de sintaxe
 - Tela de “Info”, com detalhes da VM
 - URL pública já vinculada a sua aplicação
 - Formato:
 - <https://nomeVM-usuario123456.codeanyapp.com>



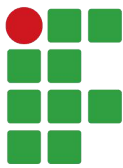
Git

- **git init**
 - Adiciona Git a um diretório local
 - Diretório passa a ser nosso repositório local
- **git config user.name "Seu Nome Bonito"**
 - Nome de usuário para aparecer no Git
- **git config user.email "aluno@email.com"**
 - E-mail para aparecer no Git
- **git ls-files**
 - Mostra arquivos que são gerenciados



Git

- **git status**
 - Mostra alterações em arquivos do diretório que precisam ser registradas no Git (commit)
- **git add [arquivo(s)]**
 - Marca um ou mais arquivos alterados para terem suas alterações registradas no Git
- **git commit -m “Mensagem obrigatória”**
 - Registra (committa) as alterações marcadas no Git, gerando uma versão

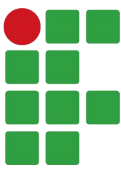


Git

- **git log**
 - Mostra log de alterações
- **git whatchanged**
 - Mostra alterações de forma mais detalhada

GitHub

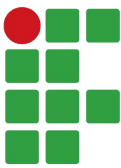
- Acesse: <https://github.com>
 - Autentique-se
- Crie um novo repositório (New Repository)
 - Escolha um nome
 - Escolha entre Private, Internal e Public
- O projeto tem uma URL pública
 - <https://github.com/usuario/projeto>
- Utilizar para clonar
 - <https://github.com/usuario/projeto.git>



Git

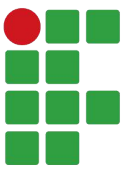
- **git remote add [alias_reposit] [uri_reposit]**
 - Vincula o repositório local (e todos os seus commits) a um repositório remoto
 - O “alias” é um apelido dado ao repositório
 - Muito comum utilizar origin
 - Exemplo:

```
git remote add origin https://github.com/usuario/projeto.git
```



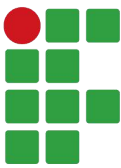
Git

- **git push -u [alias] [branch]**
 - Envia todas as alterações localmente commitadas para o repositório remoto (upstream), para uma *branch* específica
 - Exemplo:
 - **git push -u origin master**
 - A “master” é sempre a branch padrão quando ainda não temos outras criadas
 - » Normalmente ela é o fluxo de desenvolvimento principal da nossa aplicação



Git

- **git clone [uri_repositório]**
 - Usado para “baixar” todo um repositório remoto para uma pasta local, com todas as suas versões e históricos
 - Usado para baixar repositórios públicos (open source):
 - git clone <https://github.com/torvalds/linux.git>
 - Usado para colaborar com os amiguinhos:
 - git clone <https://github.com/seu-usuario-github/seu-repositorio.git> pasta-local
- ← Pasta opcional, senão usa o próprio nome do repositório como pasta



Git

- **git pull [alias] [branch]**
 - Recebe todas as atualizações do repositório remoto e mescla com o local
 - **git pull origin master**
 - Não importa se você é o criador ou clonou
 - Se alguém alterou o mesmo arquivo que você
 - Git tenta mesclar
 - Mas, às vezes, gera um conflito que deve ser resolvido manualmente no seu editor de texto