TÓPICOS AVANÇADOS EM PROGRAMAÇÃO

- Programação Web
- Introdução ao Django Admin

- Geraldino Antonio da Silva
- Curso: Computação

O que é o Django Admin?

- Veremos o que é o Django Admin e o que ele proporciona como ferramenta para criar módulos administrativos de forma rápida e extremamente fácil. Veremos, também, a aplicação final que iremos desenvolver ao longo do curso.
- Com o Django Admin, podemos, facilmente, criar uma área administrativa para gerenciar todos os nossos modelos de dados e os usuários cadastrados no site. Podemos, também, realizar todo o controle de acesso dos usuários sem programar uma única linha de código.

Por que utilizar o Django Admin?

- Simplifica as principais operações de cadastro e leitura de registros com formulários criados automaticamente.
- Gerenciamento automatizado de usuários com de acesso.

Vantagens

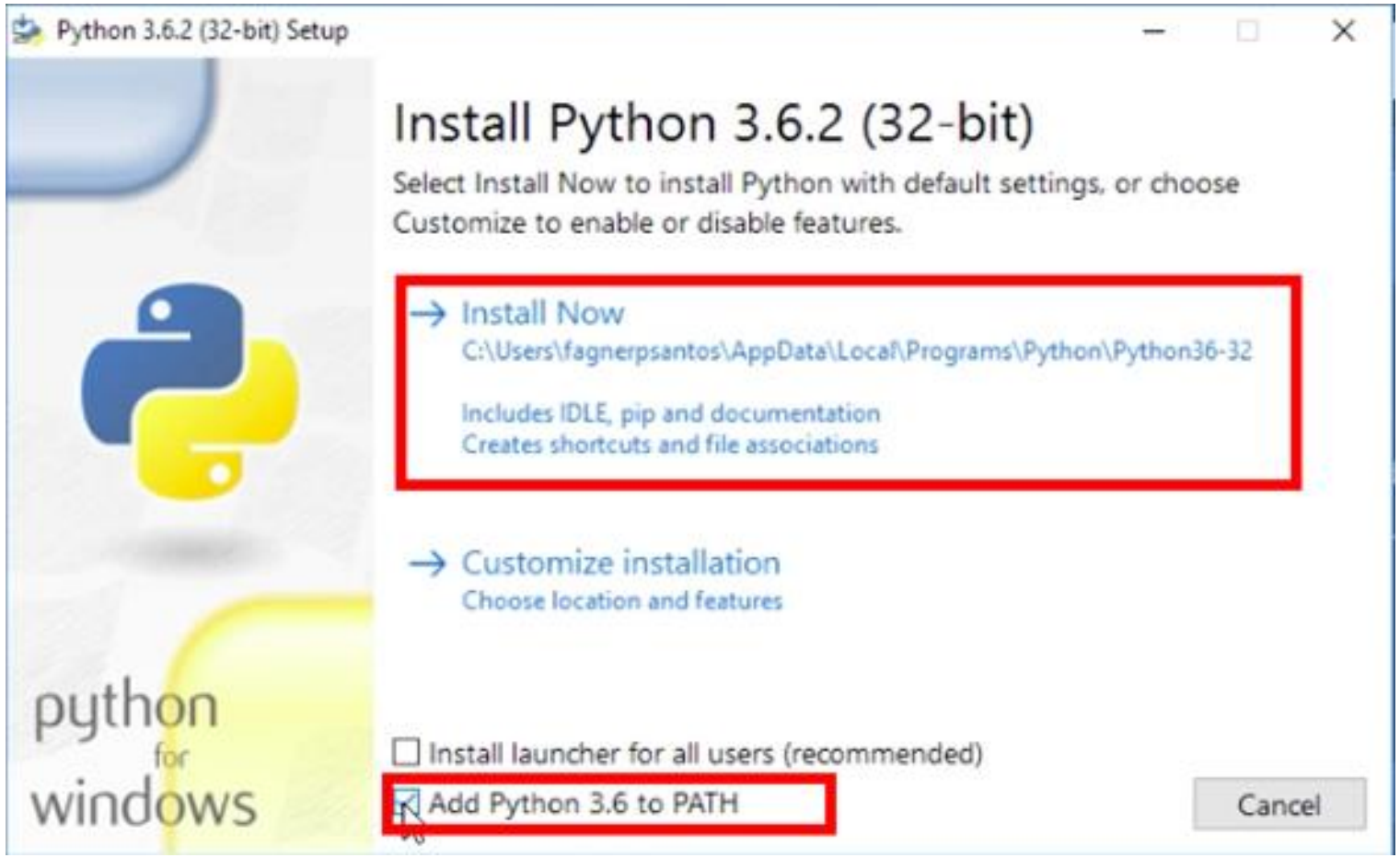
- Mapeamento Objeto-Relacional(ORM).
- Interface de administração automática.
- Criação de formulários automatizada.

Exemplo

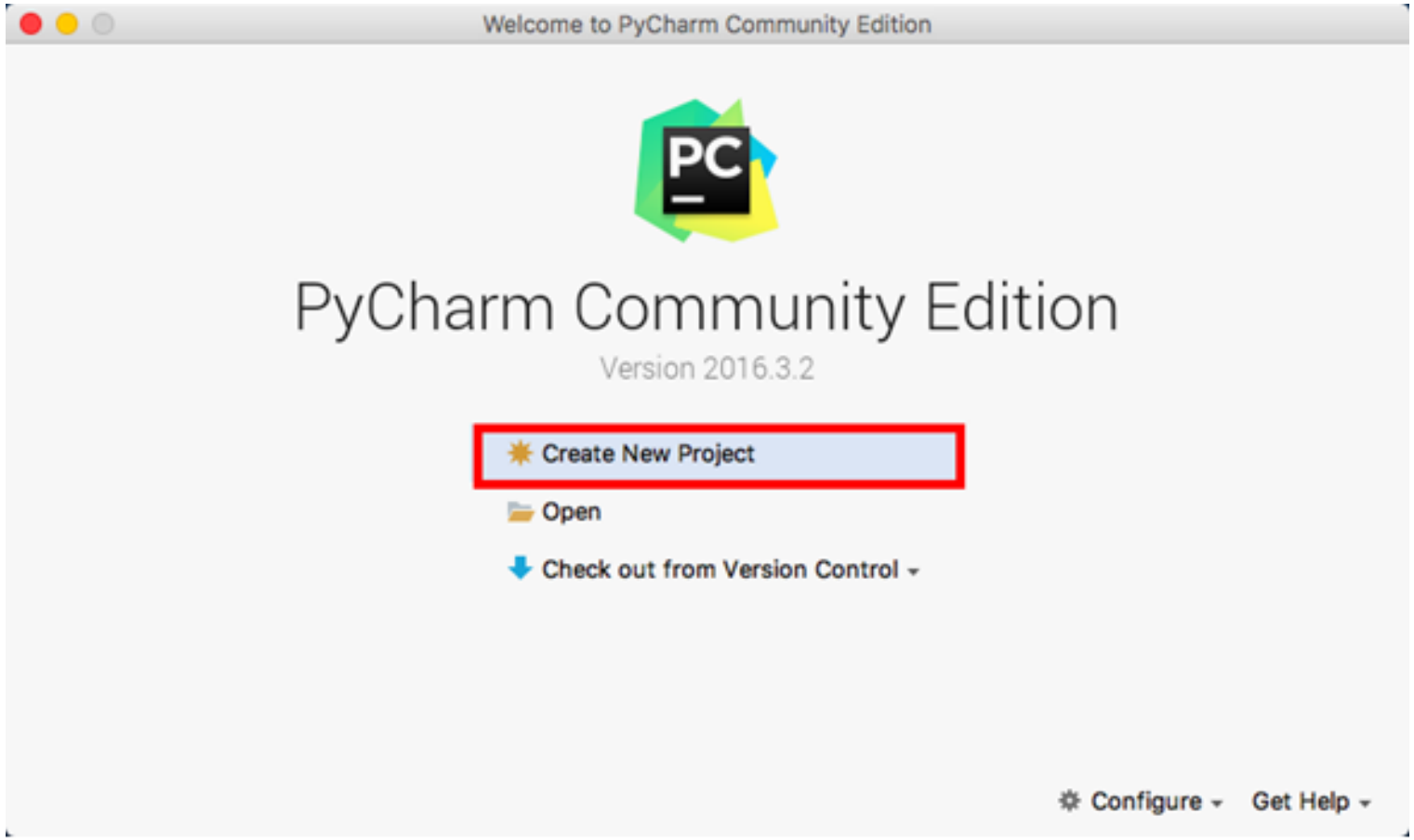
Veremos um exemplo prático de uso do Django, a fim de fixar os conceitos aprendidos na aula anterior. Aqui analisaremos uma aplicação básica para criar e listar registros, utilizando vários recursos do framework.

Download Python

Instalação Python



Instalação do Pycharm



Componentes

- **ORM** - Mapeador Objeto Relacional.
- **Template System** - Linguagem de Templates.
- **URL dispatcher** - Roteador de URLs.
- **Admin** - Interface Administrativa automatizada.
- Internacionalização.
- Gerador e validador de formulários.
- Autenticação, perfil de acessos, etc...

MVC ou MTV

- ▪**MVC** - Model View Controller.
- ▪**MTV** - Model Template View.

- **Model** - Camada responsável pela lógica de negocios, abstração de persistência etc...
- **Template** - Camada responsável pela lógica de visualização.
- **View** - Camada responsável pela interface entre Model e Template.

Ambiente (PIP)

- **PIP** – Utilitário de instalação de pacotes python.
- Baixa e instala pacotes do PyPI (Python Package Index).
- Download - https://www.python.org/download.
- Distribuições baseadas em Debian - **sudo apt-get install python-pip**.
- **pip freeze** – Lista os pacotes python instalados.
- **pip install** nome_do_pacote – Instala novos pacotes.
- **pip uninstall** nome_do_pacote – Desistala o pacote.

- **sudo apt-get install python-pip**

Ambiente (Virtualenv)

- Poderiamos desenvolver utilizando o interpretador padrão do python, mas se tivessemos 2 sistemas sendo desenvolvidos simultaneamente com versões do python diferentes?

- **virtualenv** – pacote python que tem a capacidade de criar e gerenciar ambientes isolados. Ferramenta para criar ambientes isolados de desenvolvimento, além de instalar o Django no nosso ambiente.

- **pip install virtualenv**
- **virtualenv --no-site-packages** diretorio/nome_do_ambiente
- **source** diretorio/nome_do_ambiente/bin/activate
- (nome_do_ambiente)

django-admin.py

- O modulo "**django-admin.py**" possui diversos comandos utilitários para auxliar o desenvolvedor.
- **django-admin.py help** - Lista os comandos disponíveis.
- **django-admin.py <comando>** - Definição detalhada sobre o comando.
- **django-admin.py startproject** nome_do_projeto **django-admin.py startapp** nome_da_aplicacao

Criando o projeto

- Execute o comando **django-admin.py startproject** personal_library.
- O Django criará a seguinte estrutura:
- personal_library: Pasta onde o projeto está guardado.
- personal_library: Project (Não deve ser renomeada).
- __init__.py: Arquivo vazio (indica um package).
- settings.py: Arquivo de configuração do projeto.
- urls.py: Definições de URLs do projeto.
- wsgi.py: Protocolo parecido com fastCGI serve HTTP.
- manage.py: Utilitário parecido com o **"django-admin.py".**

manage.py

- O módulo "**manage.py**" possui varios comandos utilitários.
- **syncdb** - Cria tabelas no banco de dados.
- **dumpdata --format F [aplicação]** - Extrai dados da aplicação em XML/JSON.
- **loaddata fixture** - Insere dados XML/JSON/YAML no banco de dados.
- **shell** - Interpretador Python com modelo de dados.
- **create superuser --username --email** - Cria uma usuário root.
- **runserver endereco:porta** - Inicia o servidor web de desenvolvimento.
- **startapp aplicacao** - Cria a estrutura de uma nova aplicação no projeto.

Estrutura do projeto