

CURSO DE CIÊNCIA DE DADOS APLICADA AO PODER JUDICIÁRIO



PYTHON PARA ANÁLISE DE DADOS

FICHA DO CURSO

Curso: Python para a análise de dados

Modalidade: EAD - autoinstrucional

Carga horária total: 40 horas

Carga horária semanal: 4 horas

Início da oferta:

Fim da oferta:

Pré-requisito: Recomenda-se que o cursista tenha conhecimentos em informática intermediária (instalar softwares e executar comandos em terminais) e estatística básica (média, desvio padrão, etc).

Conteudistas: Andre Ricardo Abed Grégio e Paulo Ricardo Lisboa de Almeida

1. Objetivos

Habilitar o(a) aluno(a) a utilizar a linguagem de programação Python para análise de dados. Realizar tarefas de leitura (entrada), processamento (tratamento e organização), manipulação e análise de dados com estatística descritiva e construção de gráficos simples.

Ao final do curso, o participante terá adquirido conhecimentos básicos sobre programação, será capaz de instalar e configurar seu próprio ambiente para programação e análise de dados com Python, importar e manipular dados em formatos textuais e estruturados, confeccionar e exportar gráficos, e salvar resultados de análises em arquivos.

2. Programa

Introdução ao Python; Funções; Entrada e Saída de Dados (linha de comando, argumentos, arquivos); Estruturas de dados (arrays, tuplas, listas e dicionários); Manipulação de strings; Bibliotecas/módulos padrão e suas funções; Numpy; Pandas; Visualização de Dados.

3. Procedimentos didáticos

A cada módulo (em um total de **10 módulos**) serão desenvolvidas as seguintes atividades pelos cursistas:

1. Acessar o material pré aula para preparar-se para a aula.
2. Assistir as vídeos-aulas programadas para o módulo
3. Fazer o estudo individual de materiais indicados como: leituras complementares, resolução de exercícios e acesso a vídeos adicionais.
4. Realizar as atividades de avaliação do módulo no formato de quiz.

Como suporte ao Ensino Remoto será utilizada a infraestrutura da plataforma moodle do Centro de Formação e Aperfeiçoamento de Servidores do Poder Judiciário (CEAJUD).

4. Procedimentos metodológicos

Conjunto de vídeo-aulas separadas por módulos de conhecimento acompanhado de avaliações para reforçar o aprendizado.

5. Distribuição da carga horária do cursista

A Tabela 1 a seguir apresenta distribuição da carga horária semanal total do cursista, de 4 horas por semana, nas atividades semanais previstas na vigência do curso.

Tabela 1 - Carga horária semanal do(a) cursista.

Atividade semanal	Carga horária (horas)	Fração (%)
Estudo individual de conteúdo pré-aula	00:45	18,75%
Assistir vídeo-aulas	01:15	31,25%
Estudo individual pós aula	01:00	25%
Atividades avaliativas	01:00	25%
Total	04:00	100%

6. Conteúdo programado

O curso está dividido em 10 módulos sendo que cada módulo configura uma semana de atividades. O conteúdo programado é apresentado na Tabela 2.

Tabela 2 - Cronograma detalhado do conteúdo das unidades didáticas.

Módulo	Conteúdo programado	Duração
1	<p>Ambientação ao Python</p> <p>Instalação e configuração do ambiente Python</p> <p>IDEs para trabalhar com Python</p> <p>Acesso à documentação</p> <p>Variáveis e seus tipos básicos</p> <p>Comandos de atribuição e operações básicas (soma, divisão, resto, ...)</p> <p>Realização de entrada e saída com dispositivos padrão (teclado e tela)</p>	4 horas
2	<p>Expressões lógicas: comparações de valores, operadores lógicos e estruturas de controle</p> <p>Estruturas de repetição: laços “para” (<i>for</i>) e “enquanto” (<i>while</i>).</p> <p>Funções das bibliotecas padrão (<i>pow</i>, <i>cos</i>, <i>ceil</i>, <i>sqrt</i>, ...)</p>	4 horas
3	<p>Criação de funções: parâmetros e retornos</p> <p>Estruturas de dados simples padrão: arrays, tuplas, listas</p> <p>Iteração e acesso a itens nas estruturas de dados padrão</p>	4 horas
4	<p>Estruturas de dados padrão do tipo chave-valor: dicionários</p> <p>Criação de matrizes utilizando as estruturas padrão.</p> <p>Argumentos de linha de comando (<i>argv</i>)</p>	4 horas
5	<p>Manipulação de cadeias de caracteres (<i>strings</i>)</p> <p>Entradas e saída em arquivos não estruturados</p>	4 horas

	Leitura de arquivos estruturados (csv, outros arquivos delimitados)	
6	<p>Introdução ao numpy</p> <p>Instalação do numpy no ambiente de programação</p> <p>Vetores e matrizes em numpy</p> <p>Inicialização e acesso de itens do numpy</p> <p>Importação de csv para numpy</p>	4 horas
7	<p>Operações lógicas em vetores numpy</p> <p>Ordenação de vetores</p> <p>Estatística em numpy: média, desvio padrão, máximos, mínimos, ...</p> <p>Introdução ao matplotlib</p> <p>Instalação do matplotlib</p> <p>Criando gráficos cartesianos com numpy e matplotlib.</p> <p>Criando e visualizando histogramas com numpy e matplotlib</p>	4 horas
8	<p>Introdução ao Pandas</p> <p>Instalação do Pandas</p> <p>Estruturas de dados do Pandas: Series e Dataframes</p> <p>Inicialização e acesso de itens do Pandas</p> <p>Carga de dados de arquivos csv</p> <p>Visualização de dados com auxílio do matplotlib</p>	4 horas
9	<p>Adição, remoção e alteração de colunas.</p> <p>Lidando com datas</p> <p>Rolling: médias móveis, desvio padrão, médias móveis exponenciais, ...</p> <p>Dados categóricos</p>	4 horas
10	<p>Introdução ao Seaborn</p> <p>Instalação do Seaborn</p> <p>Criando gráficos scatter e categóricos</p> <p>Criando e visualizando regressões</p>	4 horas
	Encerramento do curso	

7. Desempenho no curso

O desempenho no curso será determinado pela nota em atividade avaliativa aplicada aos cursistas. Requer-se, no mínimo, 70% de aproveitamento para obtenção de certificado.

8. Referências bibliográficas

1. Chen, Daniel Y. **Pandas for everyone: Python data analysis**. Boston: Addison-Wesley, 2018.
2. McKinney, Wes. **Python for data analysis: data wrangling with pandas, NumPy, and IPython**. Sebastopol, CA: O'Reilly Media, Inc, 2018.
3. Haslwanter, Thomas. **An introduction to statistics with Python: with applications in the life sciences**. Switzerland: Springer, 2016.
4. Dale, Kyran. **Data visualization with Python and JavaScript: scrape, clean, explore & transform your data**. Sebastopol, CA: O'Reilly Media, 2016.
5. Matthes, Eric. **Python crash course: a hands-on, project-based introduction to programming**. San Francisco, CA: No Starch Press, 2019.
6. Zumstein, Felix. **Python for Excel**. O'Reilly Media, Inc, 2021.
7. Lutz, Mark. **Programming Python**. Sebastopol, CA: O'Reilly Media, 2010.

9. Informações sobre os conteudistas

André Ricardo Abed Grégio

Bacharel em Ciência da Computação pela Universidade Estadual Paulista Júlio de Mesquita Filho (2004), Mestre em Computação Aplicada pelo Instituto Nacional de Pesquisas Espaciais (2007) e Doutor em Engenharia Elétrica pela Universidade Estadual de Campinas (2012), com período sanduíche na UCSB - University of California, Santa Barbara. Professor Adjunto do Departamento de Informática (DInf) na Universidade Federal do Paraná (UFPR). Pesquisador Colaborador do Laboratório de

Segurança e Criptografia no Instituto de Computação da Universidade Estadual de Campinas (UNICAMP). Foi Tecnologista Pleno do Centro de Tecnologia da Informação Renato Archer (CTI/MCTI) em Campinas/SP entre 2009 e 2016. Áreas e interesses de pesquisa: Segurança Computacional em geral; Comportamento de malware; Sistemas para análise de malware; Visualização, análise e mineração de dados de segurança; Segurança em SDN e IoT.

Paulo Ricardo Lisboa de Almeida

Possui graduação em Engenharia da Computação pela Universidade Estadual de Ponta Grossa (2011), Mestrado em Informática (2013) e Doutorado em Ciência da Computação (2017), ambos pela Universidade Federal do Paraná. Tem experiência em: Orientação a Objetos, Bancos de Dados Relacionais, Inteligência Computacional, Reconhecimento de Padrões, Processamento de Imagens, Sistemas Operacionais e Arquitetura de Computadores. Principais interesses de pesquisa: Reconhecimento de Padrões, Ensembles de Classificadores, Seleção Dinâmica de Classificadores e Concept Drift.

10. Requisitos Técnicos

Computador com acesso à internet. Permissão para instalar programas.

CURSO DE CIÊNCIA
DE DADOS APLICADA
AO PODER JUDICIÁRIO

