**Trabalho em Grupo – Disciplina de Novas Tecnologias (Python)**  
**Tema do Projeto:** *Sistema de Gestão de Biblioteca Digital*  
**Grupo:** 5 alunos

**Objetivo:** Desenvolver um sistema em Python que permita o gerenciamento de uma biblioteca digital simples, com cadastro de livros, usuários e empréstimos. O sistema deve usar estruturas de dados e funcionalidades baseadas nos notebooks da disciplina.

**Contexto do Projeto**

A Universidade está implementando uma Biblioteca Digital para auxiliar no empréstimo e gestão de livros em formato digital. Seu grupo foi contratado para criar a primeira versão do sistema, utilizando Python e boas práticas de programação. A aplicação será usada inicialmente para testes internos.

**Requisitos Funcionais**

O sistema deve incluir:

**1. Cadastro de Livros**

* Cada livro deve conter: título, autor, ano de publicação, ISBN e categoria.
* Utilizar dicionários e listas para armazenar os livros.
* Permitir listagem e busca por título, autor ou categoria.

**2. Cadastro de Usuários**

* Cada usuário terá: nome, e-mail, ID único e tipo (aluno, professor, visitante).
* Usar dicionários e listas.
* Evitar e-mails duplicados (usar conjuntos).

**3. Sistema de Empréstimos**

* Associar um livro a um usuário com data de empréstimo.
* Não permitir empréstimos de livros já emprestados.
* Listar todos os empréstimos ativos.

**4. Estatísticas e Relatórios**

* Quantidade de livros por categoria (uso de dicionários e conjuntos).
* Quantidade de empréstimos por tipo de usuário.
* Buscar os livros mais emprestados.

**5. Persistência de Dados**

* Salvar dados de livros, usuários e empréstimos em arquivos .txt ou .csv.
* Ao reiniciar o programa, os dados devem ser carregados novamente.

**Conteúdos Obrigatórios a Serem Utilizados**

* Atribuição e variáveis;
* Tipos numéricos e operações básicas;
* Strings (manipulação, fatiamento, formatação);
* Listas (armazenamento e manipulação);
* Dicionários (representação de dados);
* Tuplas (dados imutáveis, como tipo de usuário);
* Conjuntos (evitar duplicidades);
* Booleans e operadores lógicos (regras de empréstimo);
* Manipulação de arquivos (leitura e escrita).

**Extras (opcional, mas soma pontos!)**

* Interface textual com menu (opções para navegar no sistema).
* Funções separadas por funcionalidade.
* Comentários e tratamento de erros.

**Entrega**

* Código-fonte organizado (.py ou .ipynb);
* Arquivos gerados (.txt ou .csv);
* Relatório explicando:
  + O sistema desenvolvido;
  + O uso de cada conteúdo da disciplina;
  + Exemplos de execução;
  + Desafios enfrentados.

**Avaliação (AT1 - 2,0 pts)**