


P

# P28 - Criando uma Aplicação Completa com Flask e SQLAlchemy

Project ID: 23471781






0


2



Update README.md

LuanComputacao authored 15 hours ago

Name	Last commit	Last update
 <code>app</code>	<a href="#">Initial Setup</a>	20 hours ago
 <code>.env.example</code>	<a href="#">Initial Setup</a>	20 hours ago
 <code>.gitignore</code>	<a href="#">Initial Setup</a>	20 hours ago
 <code>README.md</code>	<a href="#">Update README.md</a>	15 hours ago
 <code>organizador_uml.png</code>	<a href="#">fix: uml</a>	15 hours ago

 `README.md`

# Postgres & Flask & Faker

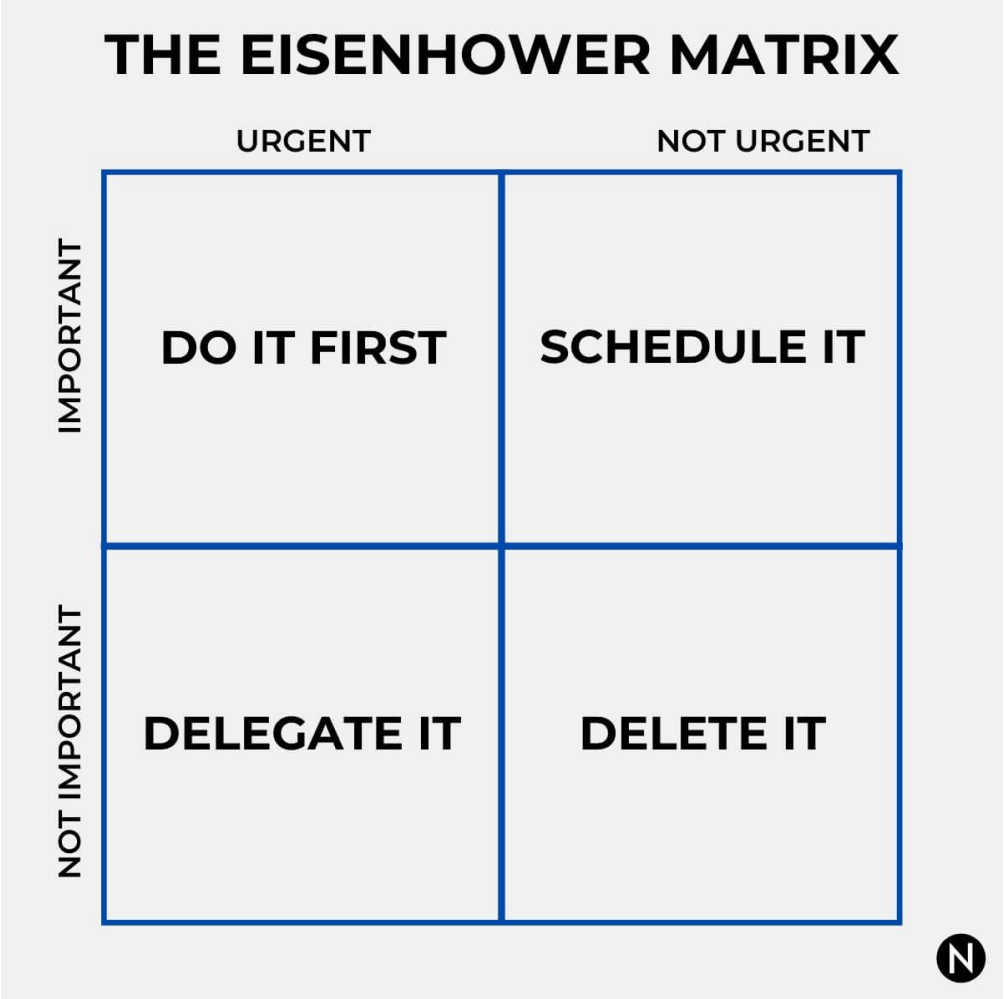
## Enunciado

Nesta atividade, você irá construir um programa que vai te auxiliar a organizar as suas tarefas do dia-dia. O programa deverá organizar as tarefas de acordo com a matriz de Eisenhower.

Para isso, você deverá utilizar **ORM SQLAlchemy** para as queries e a biblioteca **Faker** para os dados.

Mas o que é uma Matriz de Eisenhower?

É absolutamente normal recebermos uma enorme quantidade de tarefas para realizar e não ter nenhuma ideia de por onde começar. É aí que podemos fazer uso da Matriz de Eisenhower, é na verdade uma maneira simples de definir como priorizar as suas tarefas do dia a dia, de modo que, as tarefas recebem prioridades de 1 a 4, sendo elas: 1 - **"urgente e importante**, 2 - **não urgente e importante**, 3 - **não importante e urgente** e 4 - **não urgente e não importante**".



“O que é importante raramente é urgente e o que é urgente raramente é importante.” – Dwight Eisenhower, 34º Presidente Americano

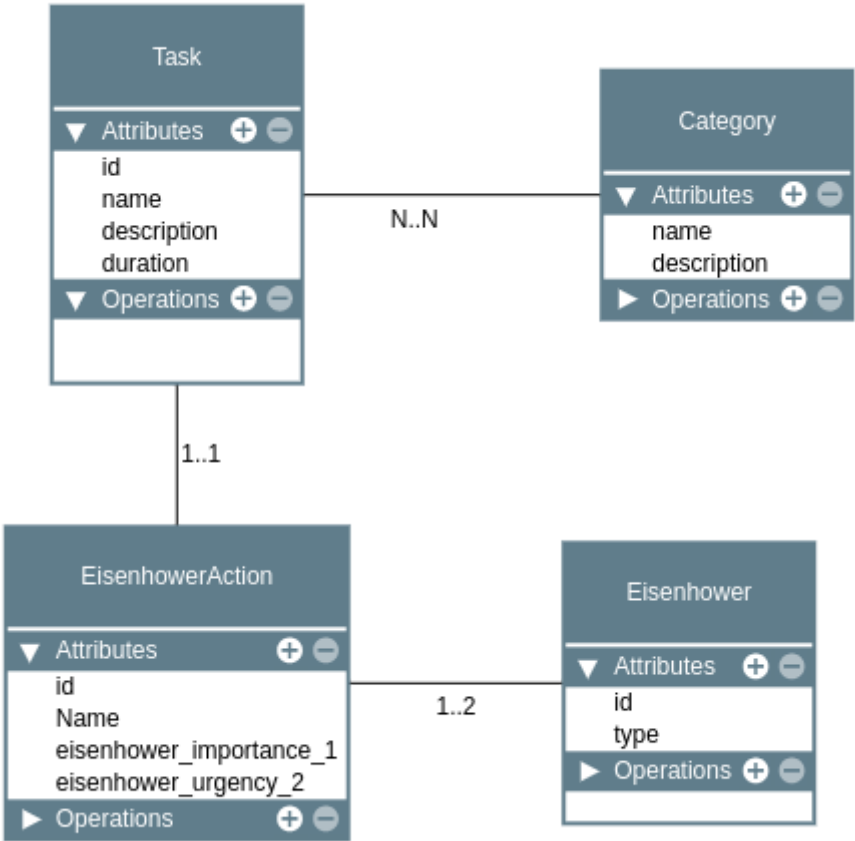
## Voltando ao enunciado

- Crie um banco de dados denominado `DatabaseTaskOrganizer`.
- Crie as seguintes tabelas:

### Categories

Antes de começar a leitura sobre as **models** veja o diagrama abaixo

- [Lik Yed Online](#)



### Category

- Deve ter como parâmetros
  - `name`
    - `VARCHAR(100)`
  - `description`.
    - `TEXT`

### Task

- Deve ter os seguintes atributos:
  - id
    - INTEGER
    - Primary Key
  - name
    - VARCHAR(100)
  - description
    - TEXT
  - duration
    - Em minutos

### Eisenhower

- Deve ter os seguintes atributos:
  - id
    - INTEGER
    - Primary Key
  - type
    - VARCHAR(100)

### EisenhowerAction

- Deve ter os seguintes atributos:
  - id
    - INTEGER
    - Primary Key
  - name
    - VARCHAR(100)
  - eisenhower\_importance\_1
    - INTEGER
  - eisenhower\_urgency\_2
    - INTEGER

Caso 1	Caso 2	Caso 3	Caso 4
eisenhower_importance_id: 1	eisenhower_importance_id: 2	eisenhower_importance_id: 1	eisenhower_importance_id: 2
eisenhower_urgency_id: 3	eisenhower_urgency_id: 4	eisenhower_urgency_id: 4	eisenhower_urgency_id: 3
Result	Result	Result	Result
'Do It First'	'Delete it'	'Delegate it'	'Schedule it'

- Crie as seguintes rotas:
  - **categories**, deve ter as seguintes rotas:

GET /categories

POST /categories  
{  
 "name": "Home Routine",

```
      "description": "djfhnjisdbnhfi"
    }
  ]
}
```

- **tasks**, deve ter as seguintes rotas:

```
GET /tasks/<category_id>
# Deve retornar uma lista de tasks de uma certa categoria incluindo a prioridade de cada task.
- Exemplo:
[
  {
    'task_name': 'Lavar louça',
    'task_description': '',
    'task_duration': 20,
    'eisenhower_priority': 'Do it first'
  }, {
    'task_name': 'Arrumar a cama',
    'task_description': '',
    'task_duration': 10,
    'eisenhower_priority': 'Delegate it'
  }, {
    'task_name': 'Estudar matemática',
    'task_description': '',
    'task_duration': 120,
    'eisenhower_priority': 'Schedule it'
  }
]
```

```
POST /tasks
{
  'category_id': '',
  'eisenhower_action_id': '',
  'task_name': '',
  'task_description': '',
  'task_duration': 0
}
```

```
DELETE /tasks/<task_id>
```

```
PATCH /tasks/<task_id>
{
  'category_id': '',
  'eisenhower_action_id': '',
  'task_name': '',
  'task_description': '',
  'task_duration': 0
}
# Pode-se atualizar qualquer atributo de uma task (um atributo ou mais) deixando-os explícitos no corpo
```

- **eisenhower\_and\_actions**, deve ter as seguintes rotas:

```
POST /eisenhower_and_actions
`data: {
  'name': 'Do It First',
  'eisenhower_importance_id': 1,
  'eisenhower_urgency_id': 3
}`
```

```
DELETE /eisenhower_and_actions/<eisenhower_actions_id>
```

## Entregáveis

### Repositório

- Código fonte
  - Commits
    - Será considerado o commit com timestamp anterior a realização da entrega no canvas
  - arquivos/módulos de código
    - do código solução
    - dos testes utilizados
  - arquivo README.md

- Privacidade
  - Privado
  - Incluir como desenvolvedores
    - peer coach
    - facilitador
    - instrutor