

# Infraestruturas e Tecnologias Organizacionais

## Licenciatura em Sistemas de Informação para Gestão

### 2º Semestre 2022/2023

#### Ficha de trabalho 8

#### Objetivos:

- Aplicações web

#### Exercícios

##### Exercício 1 – Web Application

1. Inicie um novo projeto no Pycharm. Instala a biblioteca Flask.
2. Analisa o código default ao criares o projeto Flask.

```
from flask import Flask

app = Flask(__name__)

@app.route('/')
def hello_world(): # put application's code here
    return 'Hello World!'

if __name__ == '__main__':
    app.run()
```

3. Faz run da tua primeira aplicação Web.
4. Abre no browser o endereço <http://127.0.0.1:5000>
5. Abre no browser o endereço localhost:5000
6. Modifica a mensagem a visualizares na página Web
7. Sempre que fazes uma chamada ao teu browser verifica as mensagens que vão surgindo na consola do Pycharm
8. Verifica o IP da tua máquina.
9. Modifica a seguinte linha de código e coloca o teu IP

```
app.run(host="10.180.18.92")
```

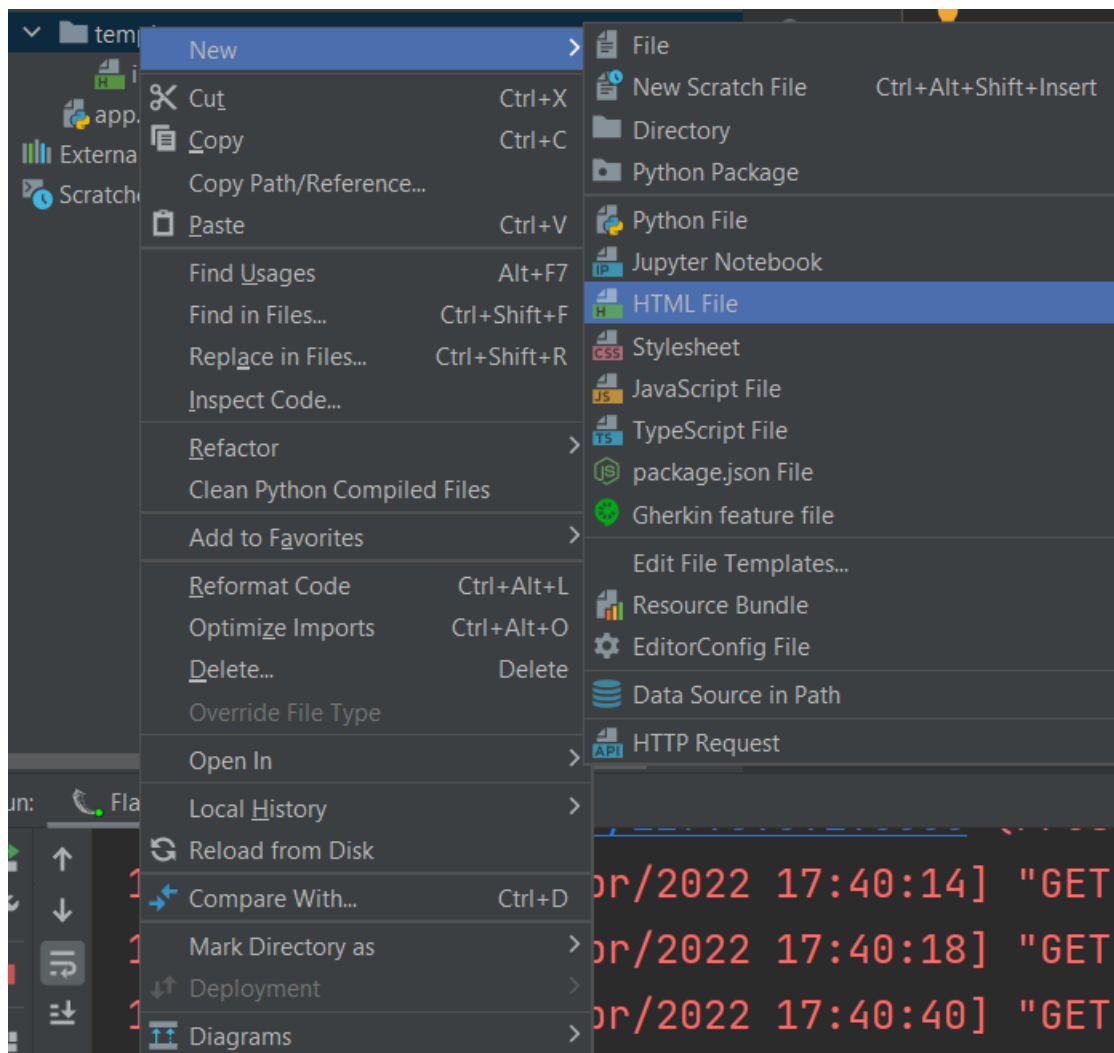
10. Pede as teus colegas que acedam ao teu serviço Web utilizando o IP:5000

## **Exercício 2 Templates**

1. Até o momento, a aplicação que criaste é simples são usar a linguagem HTML. No entanto, as aplicações Web na sua grande maioria utilizam a linguagem HTML.
2. Deste modo, vamos criar páginas HTML utilizando os templates. Acrescenta ao teu import a função `render_template`

```
from flask import Flask, render_template
```

3. Cria na pasta templates do teu projeto, um ficheiro HTML chamado index.html



4. Dentro do ficheiro HTML que criaste analisa o código.

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>My First App</title>
</head>
<body>
<h1>Welcome to Web Applications in ITO</h1>
</body>
</html>
  
```

Repara que o corpo da tua página é inserido dentro da tag <body>. A tag <h1> em HTML define o heading mais importante.

5. Modifica a mensagem a aparecer na tua página Web. Repara que para acederes à página index.html terá que acrescentar ao teu endereço /index, ou seja, <http://127.0.0.1:5000/index>
6. Testa as tags <h2>, <h3>, <h4>, <p> no teu index.html

7. No teu programa principal, acrescenta a seguinte função que chamará o ficheiro index.html que criaste

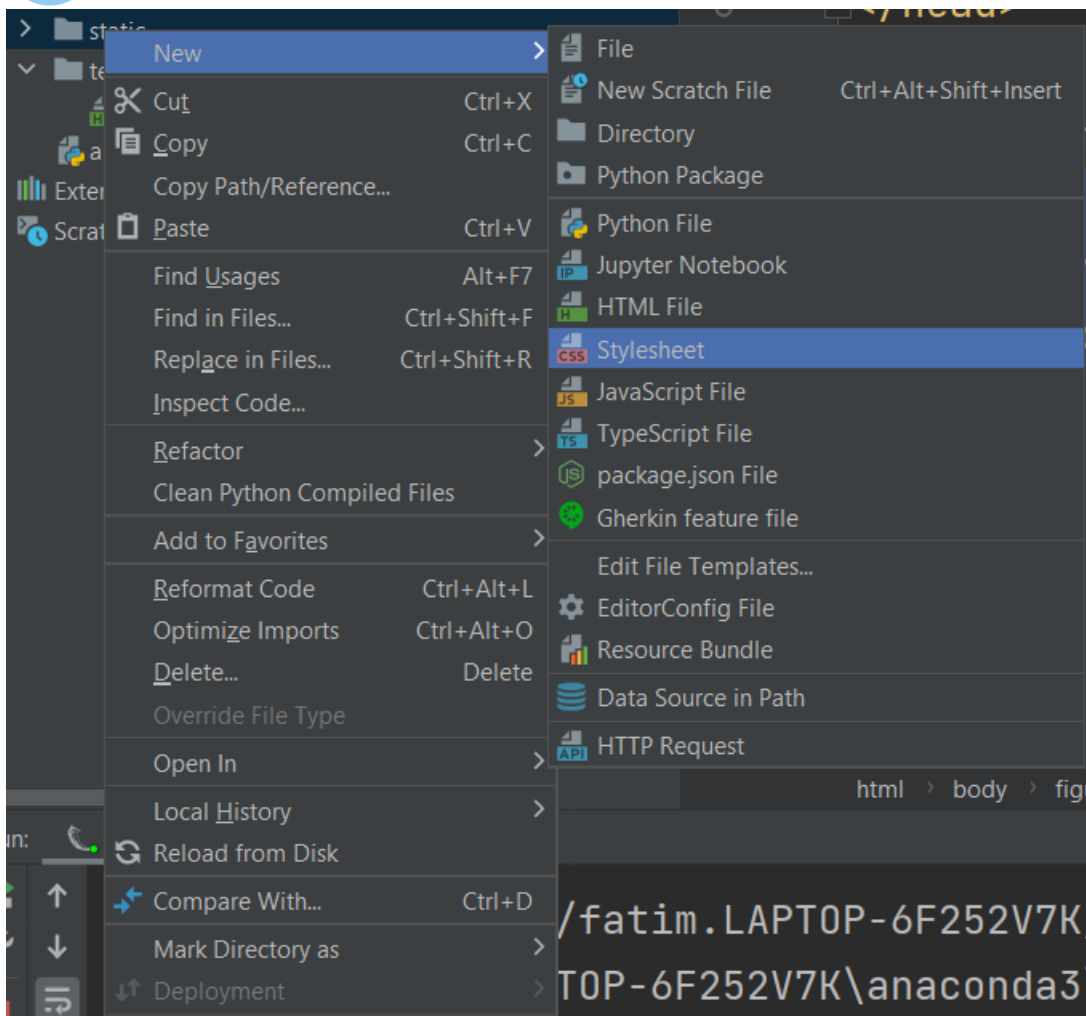
```
@app.route('/index')
def index():
    return render_template('index.html')
```

8. Faz download da imagem do Moodle, coloca-a na pasta static e acrescenta o seguinte código ao teu ficheiro index.html

```
<figure>
  
  <figcaption>Legenda da figura 1</figcaption>
</figure>
```

### Exercício 3 Ficheiros CSS

1. As aplicações Web normalmente possuem uma pasta static para ficheiros de estilo que irão formatar a tua página Web. Esses ficheiros são denominados por ficheiros CSS.
2. Na pasta static, cria um ficheiro CSS do seguinte modo:



3. Dentro do teu ficheiro CSS coloca a seguinte formatação:

```
h1 {
  border: 2px #eee solid;
  color: brown;
  text-align: center;
  padding: 10px;
}
```

4. De seguida, no head do ficheiro html onde pretendes aplicar esta formatação inclui:

```
<link rel="stylesheet" href="{{ url_for('static', filename=
'style.css') }}">
```

5. Tenta perceber o que aconteceu.
6. Cria mais páginas HTML nesta aplicação

7. Para páginas HTML explora <https://getbootstrap.com/>.
8. Cria uma nova página HTML e introduz o template de início do bootstrap e testa diferentes opções.

```

<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1">

    <!-- Bootstrap CSS -->
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.
min.css" rel="stylesheet" integrity="sha384-
1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
crossorigin="anonymous">

    <title>Hello, world!</title>
  </head>
  <body>
    <h1>Hello, world!</h1>

    <!-- Optional JavaScript; choose one of the two! -->

    <!-- Option 1: Bootstrap Bundle with Popper -->
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bu
ndle.min.js" integrity="sha384-
ka7Sk0GlN4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
crossorigin="anonymous"></script>

    <!-- Option 2: Separate Popper and Bootstrap JS -->
    <!--
    <script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.10.2/dist/umd/popper
r.min.js" integrity="sha384-
7+zCNj/IqJ95wo16oMtfKbZ9ccEh3leOz1LHGyDuCQ6wgnyJNSYdrPa03rtR1zdB"
crossorigin="anonymous"></script>
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.mi
n.js" integrity="sha384-
QJHtvGhmr9XOI6YvutG+2QOK9T+ZnN4kzFN1RtK3zEFEIsxhlmWl5/YESvpZ13"
crossorigin="anonymous"></script>
    -->
  </body>
</html>

```

9. Experimenta acrescentar um navbar
10. Acrescenta mais opções à tua escolha

## **Segundo Trabalho prático**

### **Ponto 1:**

Cria uma aplicação Web para a empresa que criaste.