

1- Google Cloud Platform

- Google Cloud Platform (GCP) e computação sem servidor (serverless)
 - <https://cloud.google.com/serverless>

Criar e implantar um serviço em Java

- <https://cloud.google.com/run/docs/quickstarts/build-and-deploy/java>
- **Nota:** o exemplo pode solicitar a ativação de uma conta com faturação

2- Google Apps Script

na categoria Workspace services

- <https://developers.google.com/apps-script/reference/>
- <https://developers.google.com/apps-script/reference/spreadsheet>
- <https://developers.google.com/apps-script/reference/drive/drive-app>
- <https://developers.google.com/apps-script/reference/mail/mail-app>
- <https://developers.google.com/apps-script/reference/language/language-app>

Notas prévias:

- sempre que o seu programa menciona um recurso adicional, terá de conceder explicitamente a autorização para esse recurso por parte do titular da conta Google que executa o programa
- se um programa demora muito tempo a executar, ou parece estar em loop, use o botão **Stop / Parar**, no mesmo painel onde iniciou a execução.

Exercício 1

- implementar um pequeno bloco de software que executará associado a uma solução cloud, cuja infraestrutura não temos de gerir

Depois de consultar os links acima, e a partir de uma nova folha de cálculo no Google Drive, crie uma pequena aplicação em AppsScript (associada à folha de cálculo),

a) que liste os ficheiros numa determinada pasta do Drive, escrevendo-os na consola;

- veja na API como aceder a uma pasta, identificando-a pelo seu id (String)
- veja como listar os ficheiros dentro dessa pasta (API para o tipo Folder)

b) criar uma função nova que tenha como argumento o identificador de uma pasta no Drive;

e deve listar e formar uma "lista ou array" os nomes dos ficheiros da pasta, devolvendo-a;

a função deve ser invocada para uma determinada pasta, os itens com os nomes devem ser registados na folha de cálculo, nas células B2 e seguintes (para baixo).

Exercício 2

A partir de uma nova folha de cálculo no Google Drive,

crie uma pequena aplicação em AppsScript (associada à folha de cálculo)

para enviar uma mensagem de correio eletrónico a cada endereço de e-mail nas células A2 a AN, parando na primeira linha em que a célula estiver vazia.

ver [exemplo no tutorial](#)

Exercício 3

Vamos agora criar um novo projeto AppScript para disponibilizar como WebApp.

<https://developers.google.com/apps-script/guides/web>

1- No seu Drive, "**Novo...** (mais...) Script do Google Apps"

Para ser uma webapp, precisa, pelo menos de uma função doGet().

Use o seguinte exemplo:

```
function doGet( request ) {
  Logger.log(JSON.stringify(request) );
  return HtmlService.createHtmlOutputFromFile("page");
}
function doPost( request ) {
  console.log(request['parameters']['nome']);
  Logger.log(JSON.stringify(request));
  return HtmlService.createHtmlOutputFromFile("page");
};
function myFunction(e) {
  Logger.log("this is a test: "+e)
};
```

2- E deve ainda adicionar um ficheiro HTML "page":

na zona de Ficheiros, adicionar (+), do tipo HTML, com o nome page (como acima).

Conteúdo inicial proposto:

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
  <h2>Teste... resultado</h2>
</body>
</html>
```

3- Para implantar (acionar o funcionamento do projeto):

Publish, ou "nova Implementação" (com o tipo **WebApp**, e disponível para si (se usar o mesmo browser como cliente) ou para Mundo/Todos)

4- Gere um novo HTML externo, no seu PC, que submeta um form para o endereço (mostrado após implantação) da sua App:

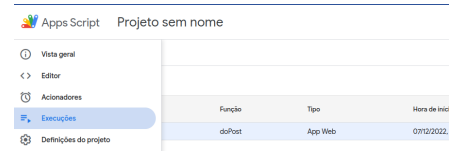
```
<html>
  <body>
    <form method="POST" action="https://script.google.com/macros/s/XXXXXXX/exec">
nome: <input type="text" name="nome" id="t1"> <br>
    <input type="submit" value="enviar">
    </form>
  </body>
</html>
```

<!-- TROCAR XXXXXXXX pelo identificador que tem (ou simplesmente troque o URL inteiro). -->

5- Teste a submissão do campo para o servidor.

6- **Ver Logs**,

ou na esquerda, separador de "Execuções"... e o detalhe de cada invocação.



O botão do form envia os campos, via HTTP POST, para a cloud, e será executado o código em doPost().

7- E Depois?

Ao tratar o processamento de doPost(), se recebe os dados poderia, por exemplo, inseri-los numa folha de cálculo do Drive (ver Spreadsheet)... (claro que já temos os GoogleForms que fazem isto muito bem)

mas vamos apenas passar dados para o documento HTML de resposta...

Crie um novo HTML, na secção de ficheiros do projeto, com o nome template:

```
<!DOCTYPE html>
<html>
  <body>
    <h2>Template com HTML e dados</h2>
    <? var data = getData(); ?>
    <table>
      <? for (var i = 0; i < data.length; i++) { ?>
        <tr>
          <? for (var j = 0; j < data[i].length; j++) { ?>
            <td><?= data[i][j] ?></td>
          <? } ?>
        </tr>
      <? } ?>
    </table>
  </body>
</html>
```

e ajuste o método doPost() para:

```
function doPost( request ) {
  dados.push(request['parameters']['nome']);
  return HtmlService.createTemplateFromFile('template')
    .evaluate();
};
var dados= [];
function getData() {
  return dados;
}
```

8 - Faça nova implantação...

e volte a testar, verificando se o valor enviado é agora mostrado no browser.

9- Mais?

Como submeter desde a linha de comandos:

```
curl -L -H 'Content-Type:application/json' -d '{"nome": "Exemplo"}'
"https://script.google.com/macros/s/Xxxxxxxx/exec"
```

Acionador / Trigger

É possível programar a execução periódica de código no script.

Veja as opções para isso, no separador "Acionadores/Triggers".

Executar operações como cliente de um serviço externo

- <https://developers.google.com/apps-script/guides/services/external>

Outra forma de implantar software (não WebApp)

Alternativa: deploy do projeto como um Serviço com API (REST) executável, que ao ser invocado remotamente vai executar código AppScript.

<https://developers.google.com/apps-script/api/how-tos/execute>

(um

, com interface ligeiramente diferente)

Outro exemplo

Um exemplo para receber dados em JSON e inseri-los numa folha de cálculo (como se fosse uma BD): [aqui](#).

✉ [Contactar suporte do site](#) ↗

Nome de utilizador: [Rodrigo Alves](#) ([Sair](#))

[Resumo da retenção de dados](#)

[Obter a Aplicação móvel](#)

Fornecido por [Moodle](#)