

- Código HTML predefinido

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5
6  body{
7      background-image: url("mars.gif");
8      background-position: center;
9      background-size: cover;
10 }
11
12 #myCanvas
13 {
14     margin-top: 30px;
15     border-width: 10px;
16     background-color: white;
17     border-style: ridge;
18 }
19
20 h1,h4
21 {
22     color: white !important;
23 }
24 </style>
25 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css">
26
27 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
28
29 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/js/bootstrap.min.js"></script>
30
31 <title>Movendo o Rover sobre Marte</title>
32 </head>

```

Style para add background

Style para Canvas

Style para tag head

Bootstrap

```

<body onload="add();">

<center>
<h1>Movendo o Rover sobre Marte</h1>
    <canvas id="myCanvas" width="800" height="600">
    </canvas>
<h4>
    <b class="text-danger">NOTA : </b> SE O ROVER NÃO ESTIVER VISÍVEL,
    PRESSIONE QUALQUER TECLA DIRECIONAL
</h4>

</center>
<script src="main.js">
</script>
</body>
</html>

```

Link com o arquivo .js

O código fornecido acima contém:

- Links bootstrap
- **style** para a tag body, adicionando fundo à tela
- **style** para o canvas, adicionando uma borda ao canvas
- **Style** para as tags h1 e h4 para criar a cor white
- Tag **Center**
- Tag **h1**, a qual contém a heading para a página
- Tag **h4**, a qual contém a nota
- **Link** para o arquivo externo JS

Adicionando evento onload

```
<body onload="add();">
```

Uma vez que a página for carregada, body é carregada e o evento onload ocorrerá.

Portanto, dentro do evento onload, colocamos uma função add. Ou seja, quando a página for carregada, a função add será chamada automaticamente.

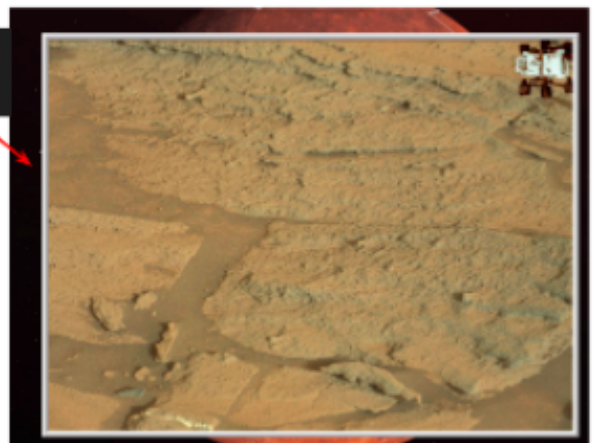
Faremos isso pois queremos que, quando a página for carregada o canvas, o rover e a imagem de fundo sejam carregadas.

Adicionando canvas -

```
<canvas id="myCanvas" width="800" height="600">  
</canvas>
```

Resultado-

```
<canvas id="myCanvas" width="800" height="600">  
</canvas>
```

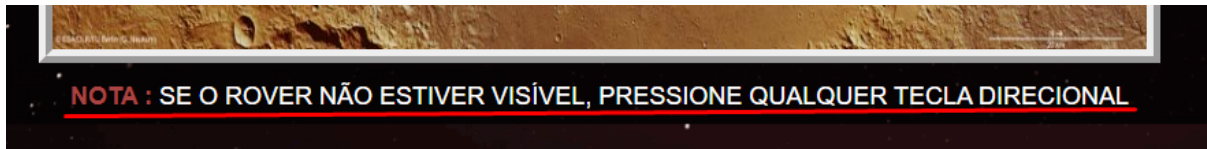


Adicionando text-danger à tag b

Código

```
<b class="text-danger">NOTA : </b> SE O ROVER NÃO ESTIVER VISÍVEL,  
PRESSIONE QUALQUER TECLA DIRECIONAL
```

Resultado -



Função add()

```
function add() {  
    backgroundImgTag = new Image(); //definindo uma variável com uma nova imagem  
    backgroundImgTag.onload = uploadBackground; // ajustando uma função, ao carregar essa variável  
    backgroundImgTag.src = backgroundImage; // carregar a imagem  
  
    roverImgTag = new Image(); //definindo uma variável com uma nova imagem  
    roverImgTag.onload = uploadrover; // ajustando uma função, ao carregar essa variável  
    roverImgTag.src = roverImage; // carregar a imagem  
}
```

Explicando drawImage()

- Sintaxe: `ctx.drawImage(img,x,y,width,height);`
- **ctx** é a referência do desenho no canvas.
 - **img**: Essa deve ser a variável image, que deve ser definida antes da definição da função.
 - **x**: em qual coordenada x a imagem deve ser desenhada
 - **y**: em qual coordenada y a imagem deve ser desenhada
 - **width** da imagem
 - **height** da imagem

Função uploadBackground()

```
function uploadBackground() {  
  ctx.drawImage(backgroundImgTag, 0, 0, canvas.width, canvas.height);  
}
```

- o **ctx** é a referência do canvas.
- o **drawImage** é a função para desenhar uma imagem no canvas.
- o **backgroundImgTag**: Essa será a variável que definimos para ajustar a imagem de fundo.
- o **0**: É a coordenada x. Ajustamos ela para 0, pois queremos que o fundo inicie de onde o canvas for iniciado.
- o **0**: É a coordenada y. Ajustamos ela para 0, pois queremos que o fundo inicie de onde o canvas for iniciado.
- o **canvas.width**: canvas.width, ou seja, **800**. Utilizamos isso, pois queremos que o fundo cubra o canvas por completo.
- o **canvas.height** - canvas.height, ou seja, **600**. Utilizamos isso, pois queremos que o fundo cubra o canvas por completo.

Função uploadrover()

```
function uploadrover() {  
  ctx.drawImage(roverImgTag, roverX, roverY, roverWidth, roverHeight);  
}
```

- o **ctx** é a referência do canvas.
- o **drawImage** é a função para chamar uma imagem no canvas.
- o **roverImgTag**: Essa será a variável que definimos para ajustar a imagem do rover.
- o **roverX**: É a coordenada x. roverX terá o valor 10, como fizemos anteriormente.
- o **roverY**: É a coordenada y. roverY terá o valor 10, como fizemos anteriormente.
- o **roverWidth**: roverWidth, ou seja, **100**, como já definimos a width do rover dentro da variável roverWidth.
- o **roverHeight**: roverHeight, ou seja, **90**, como já definimos a height do rover dentro da variável roverHeight.

Lista de caracteres e seus valores ASCII

Tecla	Valor ASCII
a	65
b	66
c	67
d	68
e	69
f	70
g	71
h	72
i	73
j	74
k	75
l	76
m	77
n	78
o	79
p	80
q	81

Tecla	Valor ASCII
r	82
s	83
t	84
u	85
v	86
w	87
x	88
y	89
z	90
Cima	38
Baixo	40
Esquerda	37
Direita	39
Alt	18
Ctl	91
Esc	27

Código para obter keypressed

```
window.addEventListener("keydown", myKeyDown);

function myKeyDown(e)
{
    keyPressed = e.keyCode;
    console.log(keyPressed);
    if(keyPressed == '38')
    {
        up();
        console.log("cima");
    }
    if(keyPressed == '40')
    {
        down();
        console.log("baixo");
    }
    if(keyPressed == '37')
    {
        left();
        console.log("esquerda");
    }
    if(keyPressed == '39')
    {
        right();
        console.log("direita");
    }
}
```