

```
using Crud.Models;
```

```
using Microsoft.AspNetCore.Mvc;
```

```
[ApiController]
```

```
[Route("api/[controller]")]
```

```
public class EventoController : ControllerBase
```

```
{
```

```
    private static List<Evento> Evento = new List<Evento>();
```

```
    private static int nextId = 1;
```

```
    [HttpPost]
```

```
    public IActionResult CriarEvento([FromBody] Evento novoEvento)
```

```
    {
```

```
        novoEvento.IdEvento = nextId++;
```

```
        Evento.Add(novoEvento);
```

```
        return CreatedAtAction(nameof(ObterEvento), new { id = novoEvento.IdAdmin },  
novoEvento);
```

```
    }
```

```
    [HttpGet]
```

```
    public IActionResult ListarEventos()
```

```
    {
```

```
        return Ok(Evento);
```

```
    }
```

```
    [HttpGet("{id}")]
```

```
    public IActionResult ObterEvento(int id)
```

```
    {
```

```
        var evento = Evento.FirstOrDefault(u => u.IdEvento == id);
```

```
        if (Evento == null)
```

```
            return NotFound();
```

```
        return Ok(Evento);
    }
}
```

```
[HttpPut("{id}")]
```

```
public IActionResult AtualizarEvento(int id, [FromBody] Evento EventoAtualizado)
```

```
{
    var evento = Evento.FirstOrDefault(u => u.IdAdmin == id);
    if (evento == null)
        return NotFound();
}
```

```
    evento.IdEvento = EventoAtualizado.IdEvento;
    evento.IdAdmin = EventoAtualizado.IdAdmin;
    evento.NomeEvento = EventoAtualizado.NomeEvento;
    evento.Preco = EventoAtualizado.Preco;
    evento.Descricao = EventoAtualizado.Descricao;
    evento.Endereco = EventoAtualizado.Endereco;
    evento.Organizadores = EventoAtualizado.Organizadores;
    return NoContent();
}
```

```
[HttpDelete("{id}")]
```

```
public IActionResult DeletarEvento(int id)
```

```
{
    var evento = Evento.FirstOrDefault(u => u.IdEvento == id);
    if (evento == null)
        return NotFound();
}
```

```
Evento.Remove(evento);
```

```
return NoContent();
```

```
}
```

