

HttpClient

Accedere alle API dal client



Accedere alle API - HttpClient

Per effettuare chiamate dal vostro client, potete utilizzare la classed .NET Core **HttpClient**

```
HttpClient client = new HttpClient();
```

Accedere alle API - HttpClient

Serve costruire una richiesta HTTP ...

```
HttpClient client = new HttpClient();

HttpRequestMessage httpRequest = new HttpRequestMessage
{
    Method = HttpMethod.Get, // HttpMethod.Post, ...
    RequestUri = fullUri
};
```

... specificando URL e HTTP Verb

Accedere alle API - HttpClient

Tramite la proprietà **Content** posso specificare il body della request ...

```
HttpClient client = new HttpClient();  
  
HttpRequestMessage httpRequest = ...  
  
Gift newGift = new Gift();  
string json = JsonConvert.SerializeObject(newGift);  
httpRequest.Content = new StringContent(json,  
    Encoding.UTF8, "application/json");
```

... lo devo specificare serializzandolo come JSON.

Accedere alle API - HttpClient

Tramite il metodo **SendAsync()** vado ad invocare la mia API ...

```
HttpClient client = new HttpClient();

HttpRequestMessage httpRequest = //...
string json = //...
httpRequest.Content = //...

var response = await client.SendAsync(httpRequest);

if (response.IsSuccessStatusCode)
{
    // ...
}
```

... l'esito lo leggo tramite la proprietà **IsSuccessStatusCode** della risposta

Accedere alle API - HttpClient

Posso accedere al body della Response, che sarà in JSON ...

```
HttpClient client = new HttpClient();

HttpRequestMessage httpRequest = //...
string json = //...
httpRequest.Content = //...
var response = //...

if (response.IsSuccessStatusCode)
{
    var jsonResponse = await response.Content.ReadAsStringAsync();
    var result = JsonConvert.DeserializeObject<GiftContract>(jsonResponse);
}
```

... e quindi devo de-serializzarlo
in un oggetto C#