TREINAMENTO .NET/REST API

TÓPICOS DE ABORDAGEM

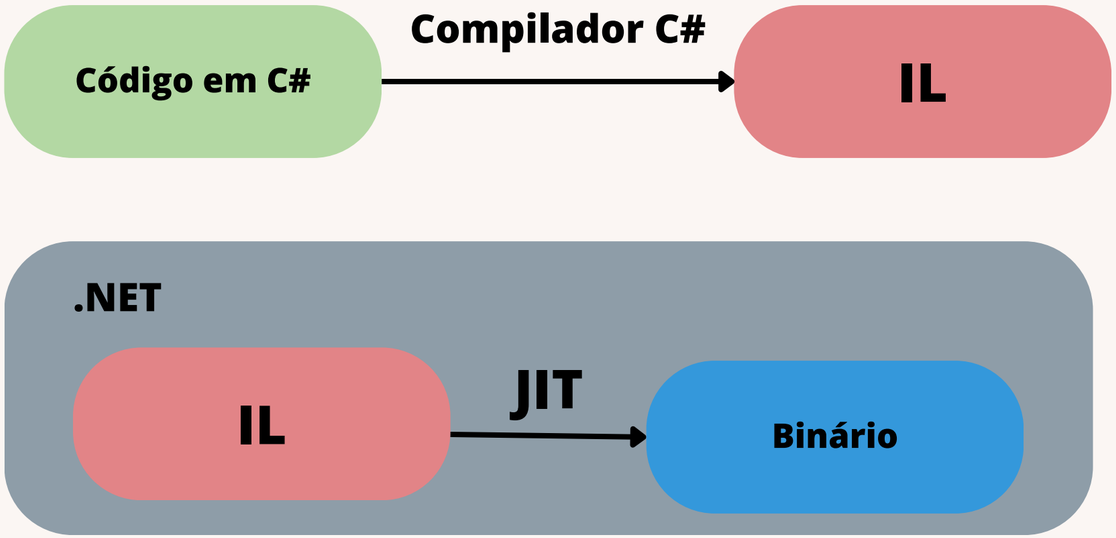
- Ecossistema .NET
- REST API e HTTP
- Começa a programar uma API de escolas
  - `dotnet new webapi`
- Parâmetros de rotas
- Parâmetros de query
- Atributos para validação

ARQUITETURA .NET

- Interoperabilidade;

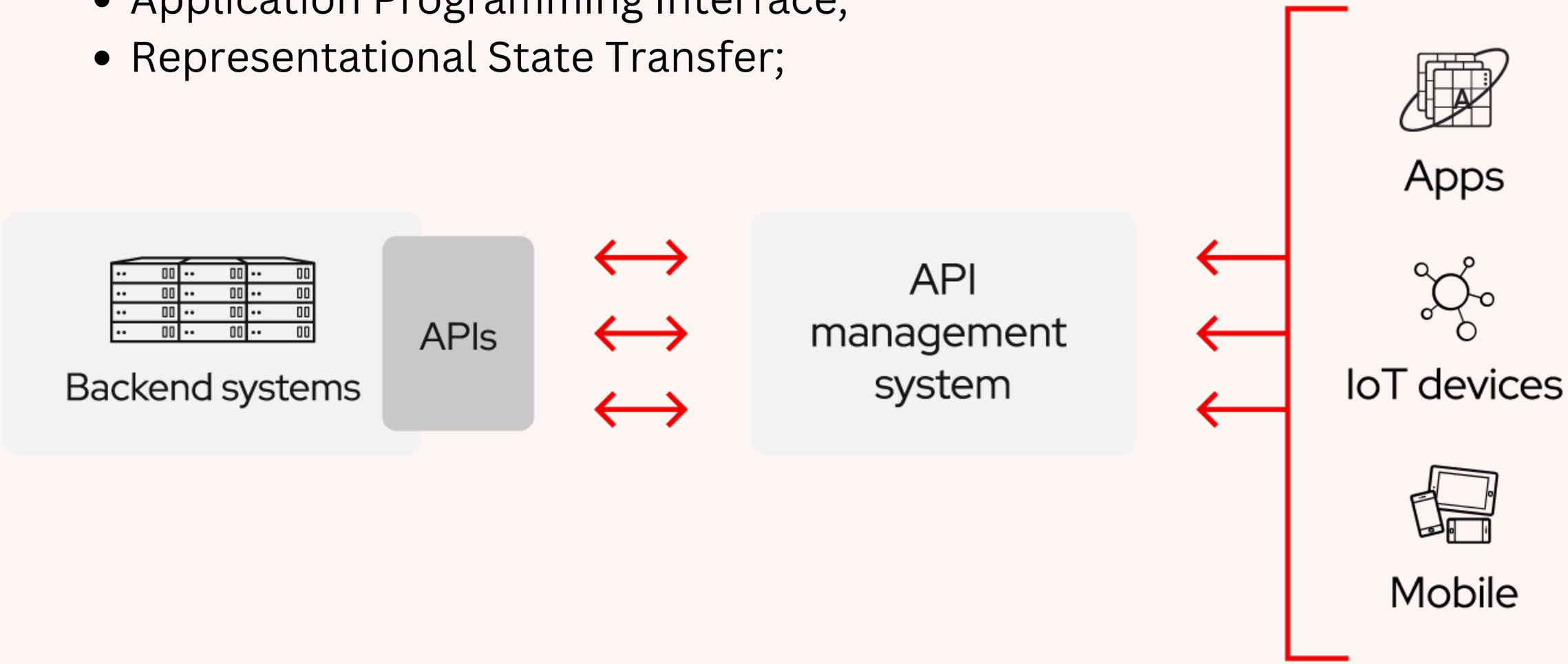
MODELO NÃO GERENCIADO

MODELO GERENCIADO



REST API

- Application Programming Interface;
- Representational State Transfer;



HTTP REPL

```
(Disconnected)> connect http://localhost:5004 --swagger /swagger/v1/swagger.json
Checking http://localhost:5004/swagger/v1/swagger.json... Found
Parsing... Successful

Using a base address of http://localhost:5004/
Using OpenAPI description at http://localhost:5004/swagger/v1/swagger.json
For detailed tool info, see https://aka.ms/http-repl-doc

http://localhost:5004/> ls
.                []
Pizza            [GET|POST]
WeatherForecast  [GET]

http://localhost:5004/> cd Pizza
/Pizza    [GET|POST]

http://localhost:5004/Pizza> ls
.       [GET|POST]
..      []
{id}    [GET|PUT|DELETE]
{num}   []

http://localhost:5004/Pizza> ls -r
.              [GET|POST]
..             []
{id}           [GET|PUT|DELETE]
{num}          []
    {denom}    [GET]
```

```
http://localhost:5004/Pizza> get 1/3
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Date: Tue, 26 Sep 2023 15:46:10 GMT
Server: Kestrel
Transfer-Encoding: chunk

{
  "result": 0.33333334
}
```

```
dotnet tool install --global Microsoft.dotnet-httprepl
```

COISAS WEB

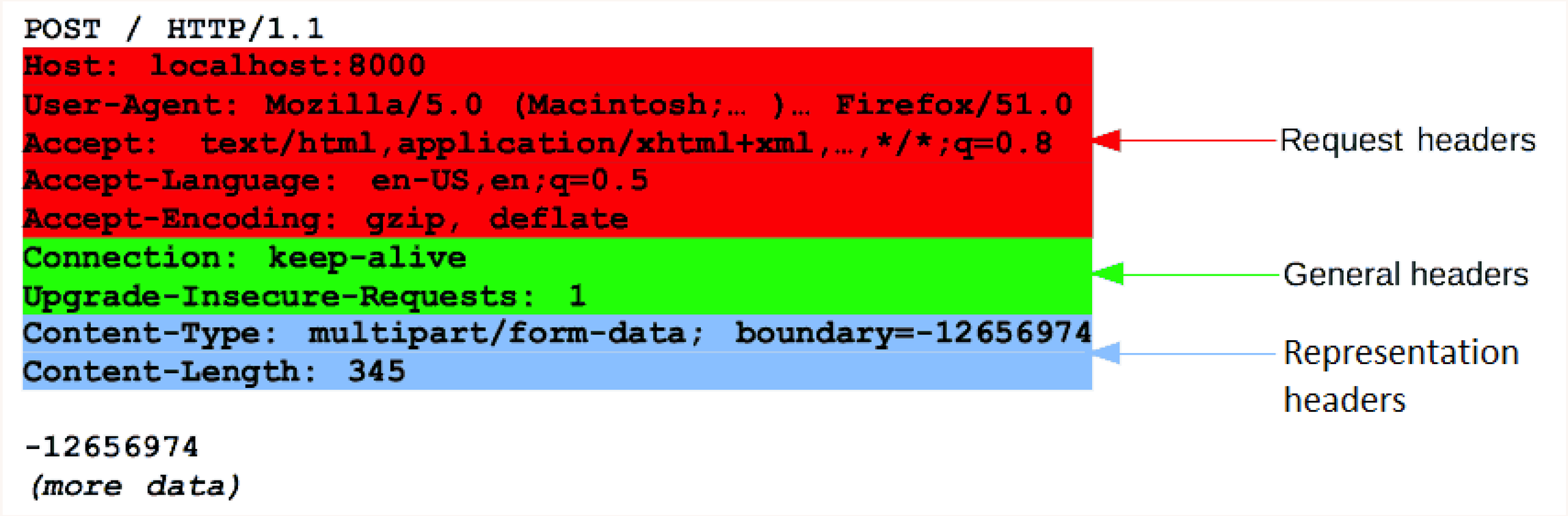
MDN | HTTP Messages

COISAS WEB
HTTP

Métodos/verbos HTTP:

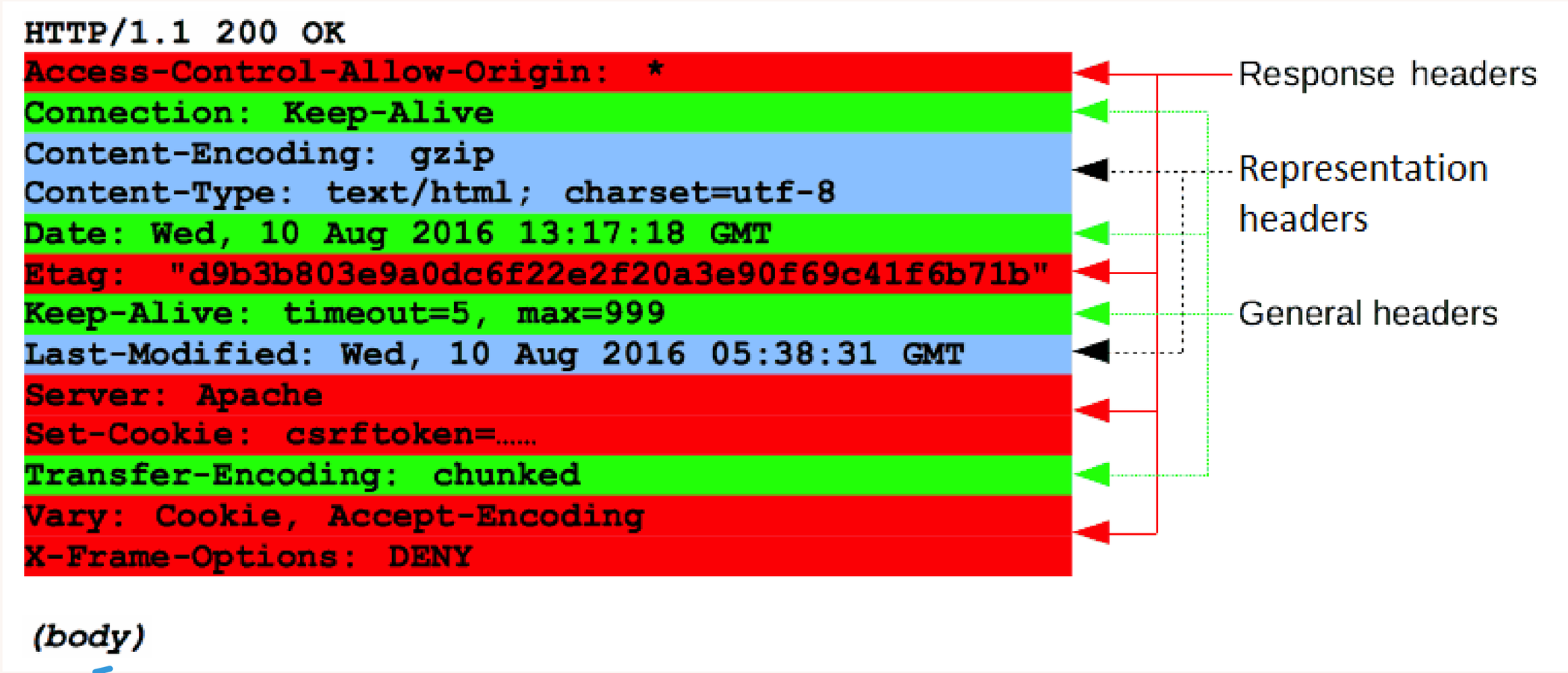
GET
POST
PUT
PATCH
DELETE
OPTIONS
HEAD

COISAS WEB
REQUISIÇÃO HTTP



O que mais pode estar tem em uma requisição HTTP?

COISAS WEB
RESPOSTA HTTP



O que vem no body de uma resposta HTTP?

COISAS WEB
PARÂMETROS HTTP

Exemplos de requisições:

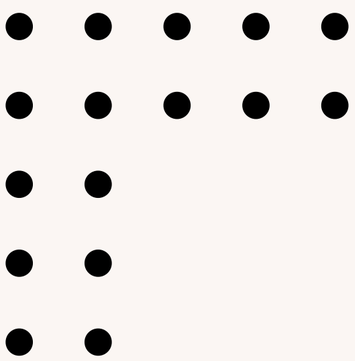
```
GET     /background.png
OPTIONS /anypage.html
GET     /accounts/me
POST    /user
```

Parâmetros de rota

? query params

```
GET  https://github.com/yudi-azvd/
GET  /products?page=10&marca=samsung&preco=gte.100
HEAD /test.html?query=alibaba
GET  https://www.youtube.com/watch?v=fcZXfoB2f70
```

OBRIGADO



PADRÕES DE PROJETO

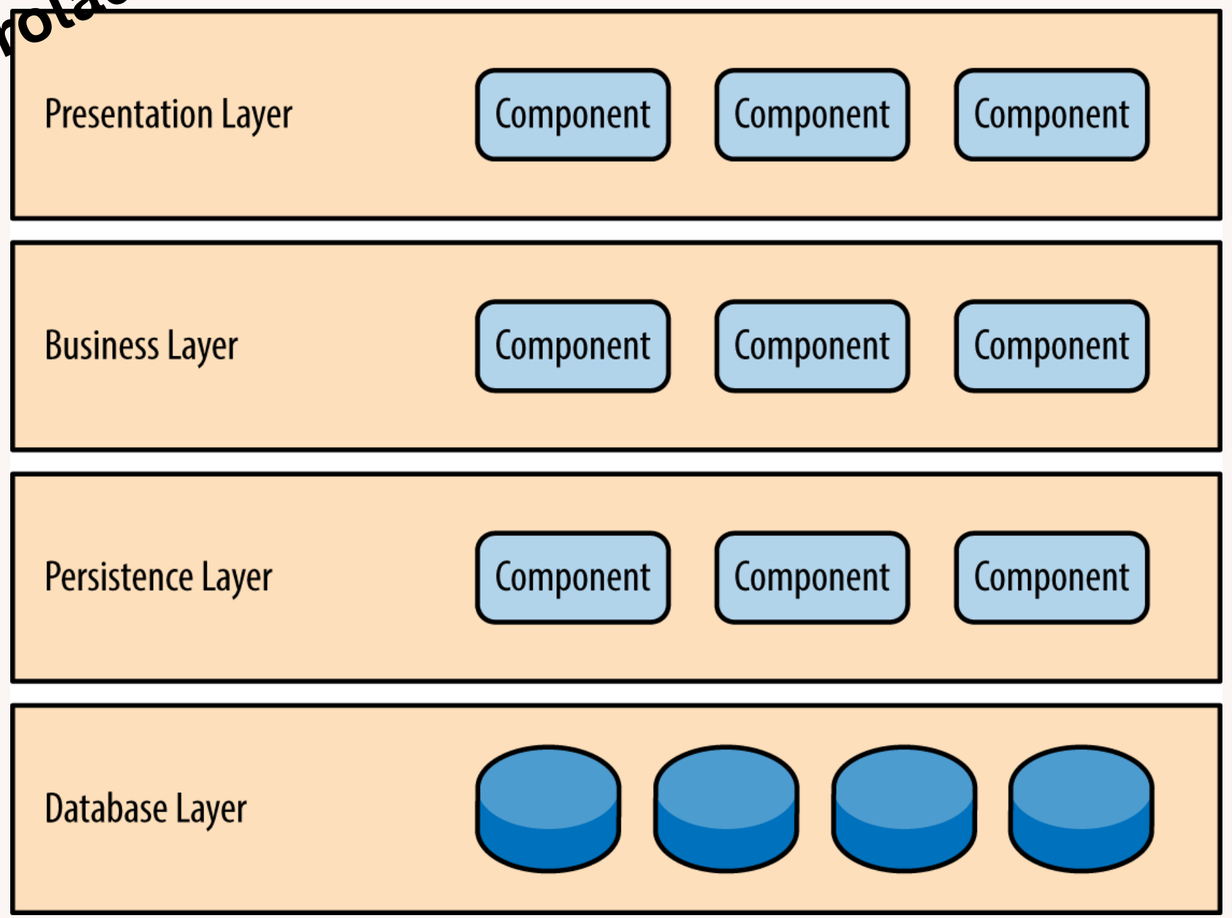
PADRÕES DE PROJETO: CONTROLADORA

Responsável por responder requisições

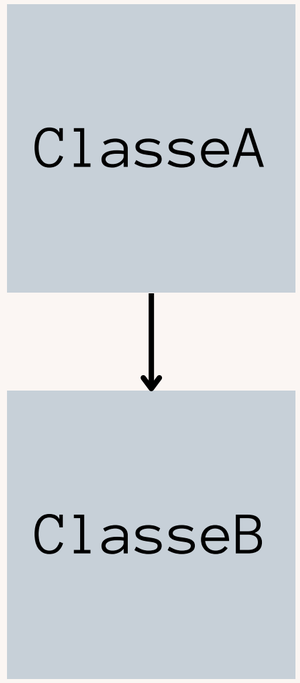
PADRÕES DE PROJETO: SERVIÇO

Controladora?

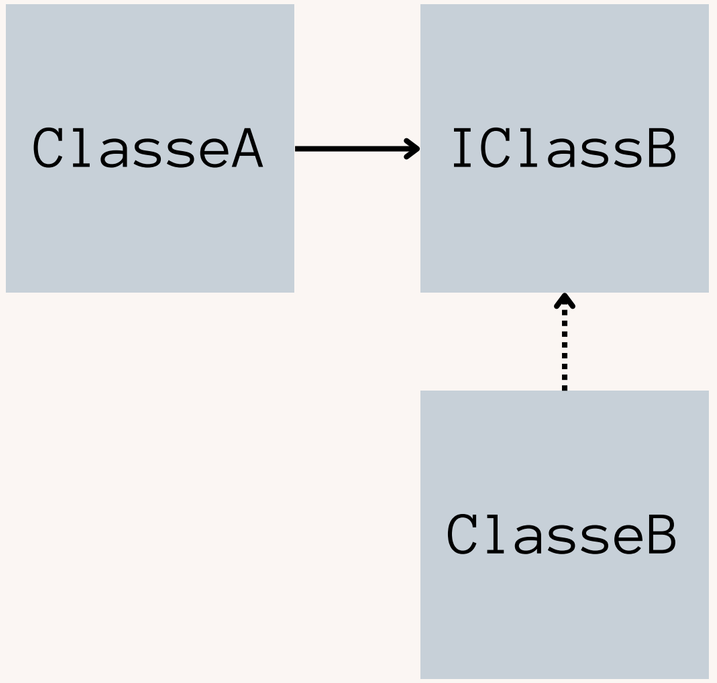
Serviço →



PADRÕES DE PROJETO: INVERSÃO DE DEPENDÊNCIA



Dependência tradicional



Inversão de dependência

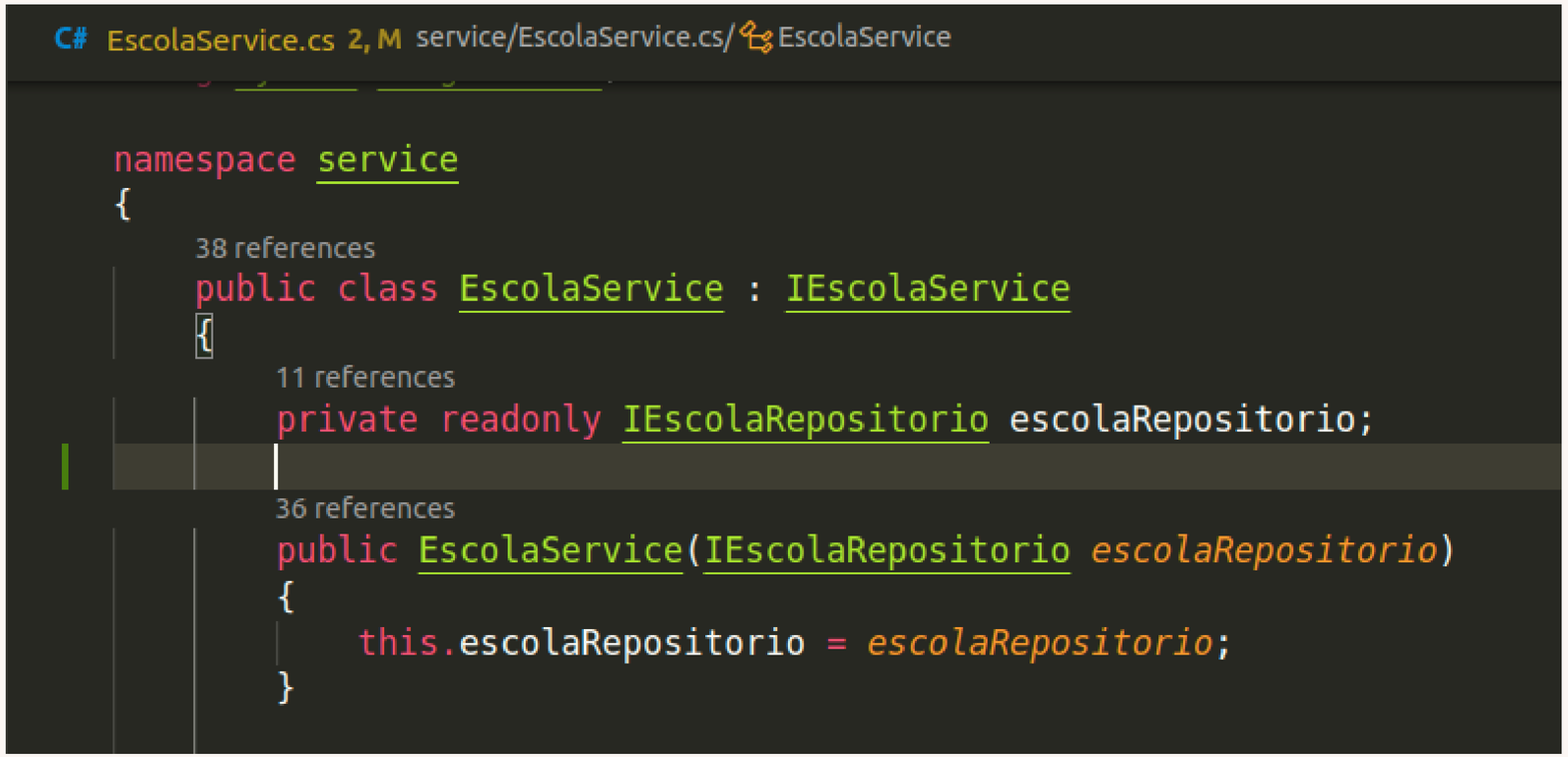
PADRÕES DE PROJETO: INVERSÃO DE DEPENDÊNCIA

Menu.java src/Menu.java

```java
import java.util.Scanner;

import controladoras.ControladoraAcesso;
import controladoras.ControladoraCadastro;
import controladoras.ControladoraPagamento;
import excecoes.OpcaoInvalidaException;

public class Menu {
    Scanner entradaDoTeclado = new Scanner(System.in);
    ControladoraCadastro controladoraCadastro = new ControladoraCadastro(ent
    ControladoraPagamento controladoraPagamento = new ControladoraPagamento(
    ControladoraAcesso controladoraAcesso = new ControladoraAcesso(entradaDo

    public void executa() {
        int opcao = -1;

        do {
            try {
                mostrarOpcoes();
```

Dependência tradicional

PADRÕES DE PROJETO: INVERSÃO DE DEPENDÊNCIA



Inversão de dependência

Dnit-EscolaService

PADRÕES DE PROJETO: REPOSITÓRIO

Persistência de dados

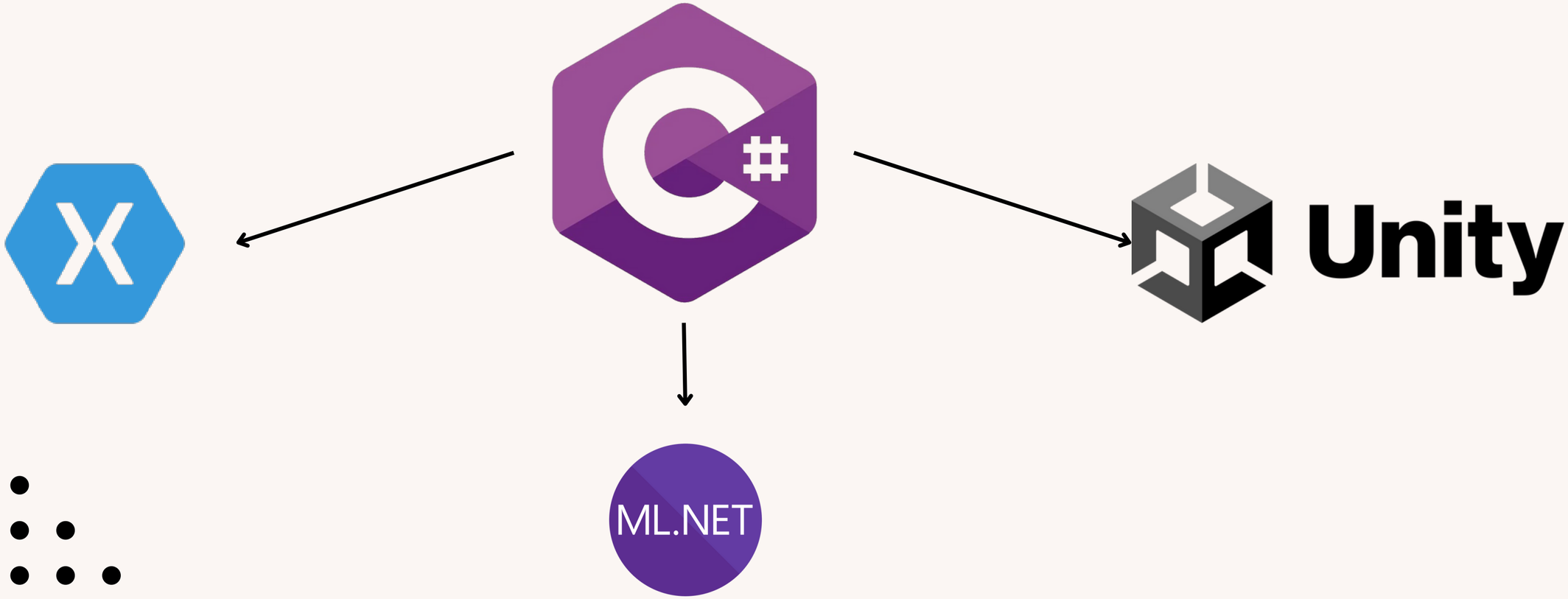
Arquivos?
NoSQL?
SQL?
ORM?

HISTÓRIA

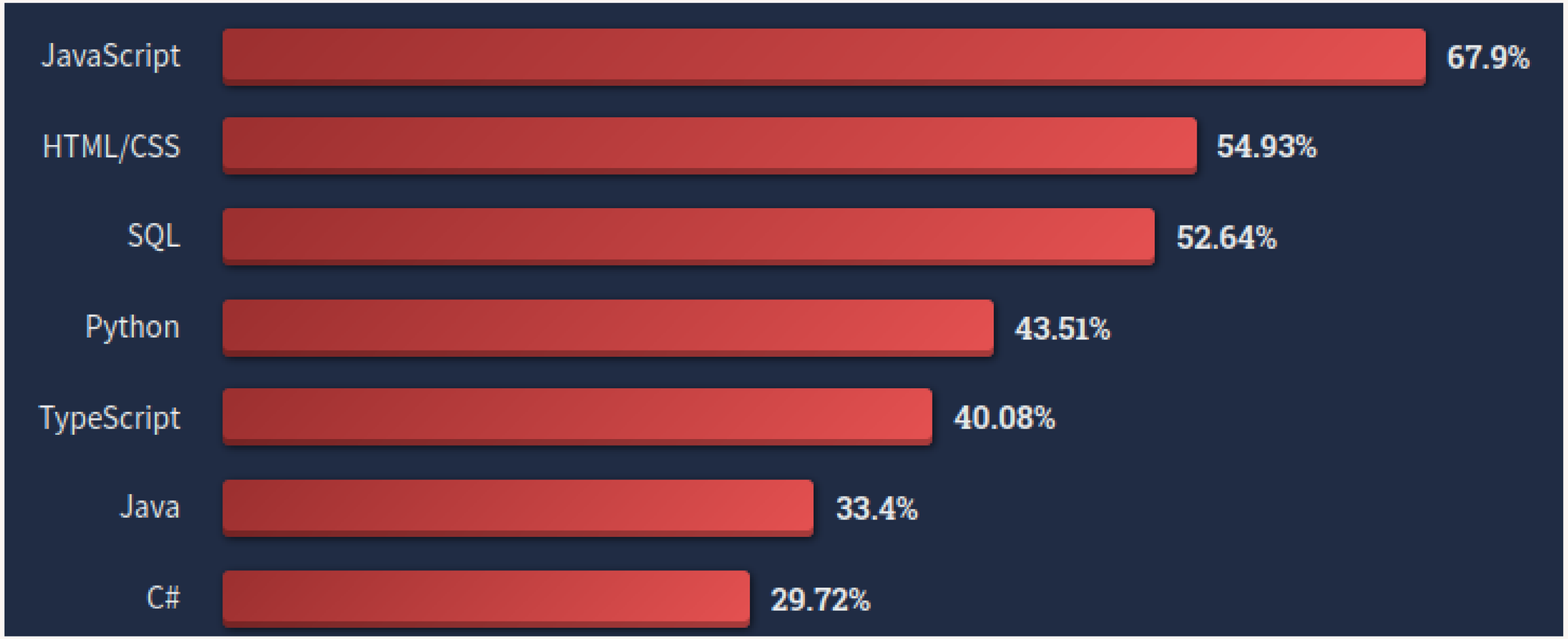
- Cool;
- Lançado pela Microsoft em 2000;
- Fortes Princípios de OO;
- Influência de Java e C++;
- Cross-Platform e Open Source

HISTÓRIA

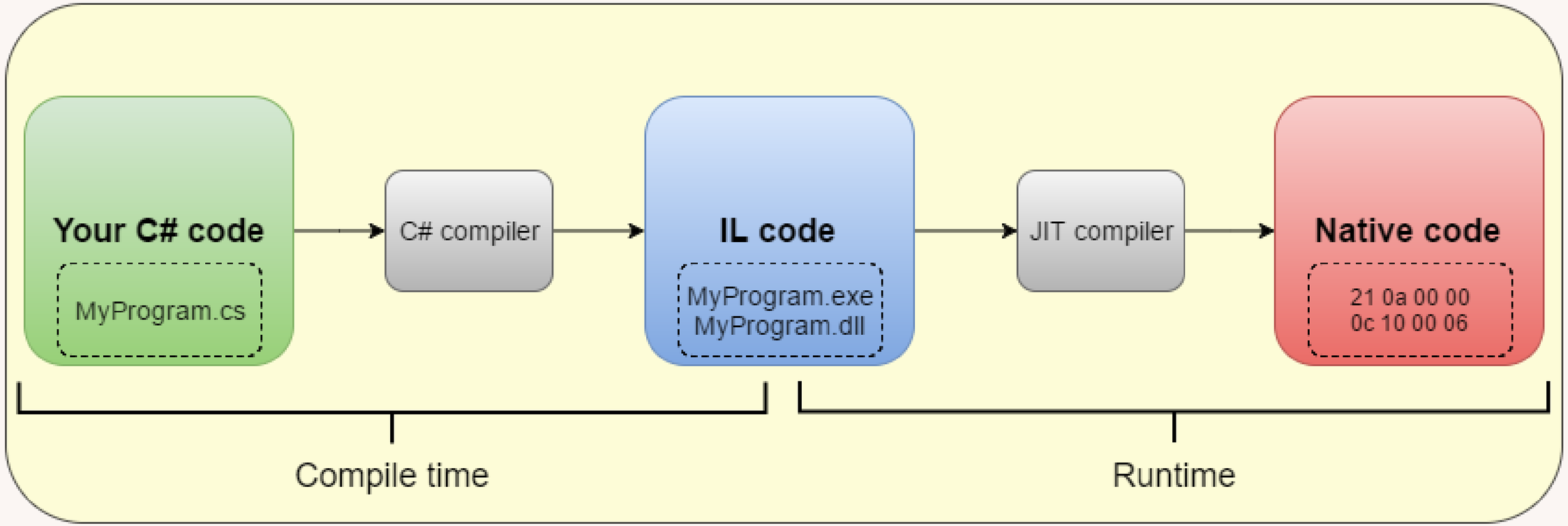
VERSATILIDADE



HISTÓRIA


JavaScript
67.9%
HTML/CSS
54.93%
SQL
52.64%
Python
43.51%
TypeScript
40.08%
Java
33.4%
C#
29.72%

PROCESSO DE COMPILAÇÃO



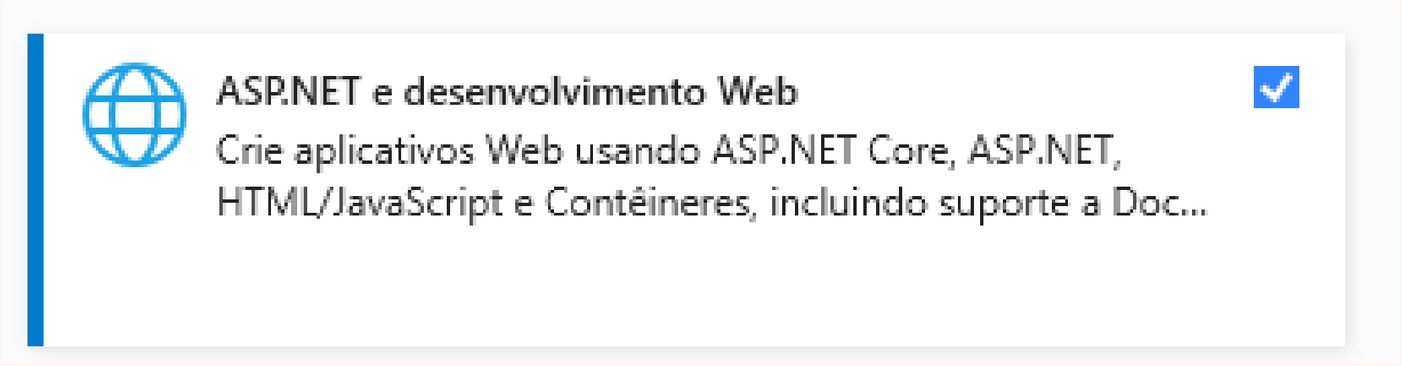
INSTALAÇÃO

INSTALAÇÃO – WINDOWS

1

Baixe o Visual Studio Community

2



No instalador, marque a opção ASP.NET e desenvolvimento Web

3

Instalar

TIPOS DE DADOS PRIMITIVOS

Tipo	Tamanho (bytes)	
int	4	int myNum = 5;
float	4	float myFloatNum = 5.99D;
double	8	double myDoubleNum = 5.99D;
char	1	char myLetter = 'D'
string	2 por caracter	string myText = "Hello"; string myText = $"Soma 1 + 2 = {1 + 2}";
bool	1	bool myBool = true; bool myBool = false;

ARRAYS

```
string[] cars = new string[4] {"Volvo", "BMW", "Ford", "Mazda"};
string[] cars = new string[] {"Volvo", "BMW", "Ford", "Mazda"};
```

```
int[] myNum = {10, 20, 30, 40};
int[] myNum = new int[]{10, 20, 30, 40};
```

LISTAS

```
List<int> numeros = new List<int> { 1, 2, 3, 4, 5, 6 };
var numeros = new List<int> { 1, 2, 3, 4, 5, 6 }
```

Adição de elementos	Append()	numeros.Append(10);
Maior elemento	Max()	var maior = numeros.Max();
Soma dos elementos	Sum()	var soma = numeros.Sum();

Conversão dos elementos	ConvertAll()
	Select()
		var dobro = numeros.ConvertAll(x => x * 2);
		var triplo = numeros.Select(x => x * 3).ToList();
		List<string> numerosString = numeros.ConvertAll(x => x.ToString());

ORIENTAÇÃO A OBJETOS

VEÍCULOS