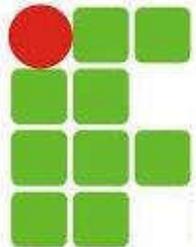
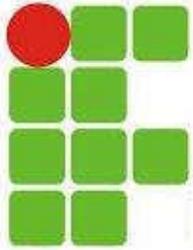

ALGORITMOS

Professor: Diego Oliveira



**Conteúdo 16:
Métodos**





Métodos

- Até agora, todo o código desenvolvido ficou dentro do método main()
- Porém pode-se criar quantos métodos forem necessários para resolver um problema
- O exemplo mais clássico é a calculadora com as 4 operações básicas:
 - soma()
 - subtracao()
 - multiplicacao()
 - divisao





Exemplos de Métodos

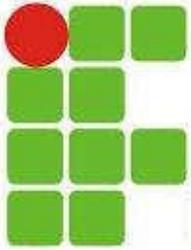
```
public class Calculadora {
    static int x = 10;
    static int y = 20;
    public static void main(String[] args) {
        soma();
        subtracao();
        multiplicacao();
        divisao();
    }
    public static void soma() {
        System.out.println("X+Y="+(x+y));
    }
    public static void subtracao() {
        System.out.println("X+Y="+(x-y));
    }
    public static void multiplicacao() {
        System.out.println("X+Y="+(x*y));
    }
    public static void divisao() {
        System.out.println("X+Y="+(x/y));
    }
}
```

SAÍDA

Output - CA (run) X

```
run:
X+Y=30
X+Y=-10
X+Y=200
X+Y=0
```



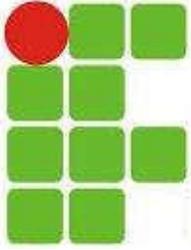


Métodos

- No slide anterior temos 5 setas
 -  o modificador de acesso **public** permite que o método seja utilizado por outras classes (assunto de POO)
 -  o tipo de retorno **void** significa que este método não retorna nada. Pode ser trocado por **int**, **double**, **String**, **boolean**...
 -  **soma** representa o nome do método
 -  os **parênteses vazios** significam que este método não recebe parâmetros, ou seja, valores de quaisquer tipos
 -  mostra que as VARIÁVEIS podem ser criadas fora dos métodos, assim elas terão um escopo GLOBAL, ou seja, podem ser acessadas de todos os métodos



- Observe que os métodos são chamados dentro do MAIN para serem executados! 4



Métodos

- Observe que os 4 métodos criados foram chamados dentro do método **main()**
- Observe também que as variáveis X e Y foram criadas como **static** para serem usadas nos métodos (em POO será diferente)
- Os 4 métodos são **void**, não retornam nada
- Pode-se estabelecer um tipo de retorno, como **int**, e fazer a impressão no próprio MAIN vejamos:





Retorno de Métodos

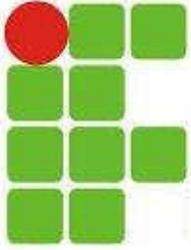
```
public class Calculadora2 {  
    static int x = 50;  
    static int y = 10;  
    public static void main(String[] args) {  
        System.out.println("X+Y=" + soma());  
        System.out.println("X-Y=" + subtracao());  
        System.out.println("X*Y=" + multiplicacao());  
        System.out.println("X/Y=" + divisao());  
    }  
    public static int soma() {  
        return x+y;  
    }  
    public static int subtracao() {  
        return x-y;  
    }  
    public static int multiplicacao() {  
        return x*y;  
    }  
    public static int divisao() {  
        return x/y;  
    }  
}
```

SAÍDA

Output - CA (run) X

```
run :  
X+Y=60  
X-Y=40  
X*Y=500  
X/Y=5
```

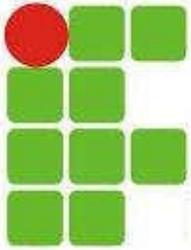




Métodos

- No segundo exemplo mostrado, vimos que é possível chamar os métodos e obter um retorno destes métodos
- Porém este retorno é sempre o mesmo, pois os valores são fixos (X e Y)
- E se quisermos efetuar os cálculos com os nossos próprios valores?
- Para isso utilizaremos os parâmetros!





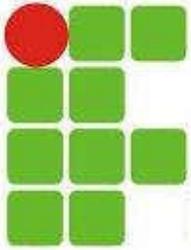
Parâmetros de Métodos

```
public class Calculadora3 {  
    public static void main(String[] args) {  
        Scanner leia = new Scanner(System.in);  
        System.out.println("Digite o valor de X: ");  
        int x = leia.nextInt();  
        System.out.println("Digite o valor de Y: ");  
        int y = leia.nextInt();  
        System.out.println("X+Y=" + soma(x,y));  
        System.out.println("X-Y=" + subtracao(x,y));  
        System.out.println("X*Y=" + multiplicacao(x,y));  
        System.out.println("X/Y=" + divisao(x,y));  
    }  
    public static int soma(int x, int y){  
        return x+y;  
    }  
    public static int subtracao(int x, int y){  
        return x-y;  
    }  
    public static int multiplicacao(int x, int y){  
        return x*y;  
    }  
    public static int divisao(int x, int y){  
        return x/y;  
    }  
}
```

SAÍDA

```
Output - CA (run) × HTTP Server  
run:  
Digite o valor de X:  
100  
Digite o valor de Y:  
10  
X+Y=110  
X-Y=90  
X*Y=1000  
X/Y=10
```

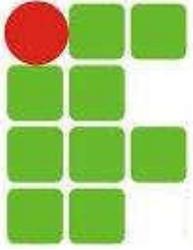




Exercício

- Escreva um método que imprima uma matriz de zeros de X por Y, recebidos por parâmetro. Este método **não retorna nada**.
- Escreva um método que receba um número e **retorne** uma String dizendo se o número é PAR ou ÍMPAR
- Escreva um método que receba 3 números inteiros X, Y e Z e **retorne** qual o maior deles
- Dentro do método MAIN, imprima um MENU que permitirá ao **usuário escolher** qual dos 3 métodos deseja chamar. Inclua uma opção de SAIR.
- O algoritmo deverá ficar rodando até que o usuário escolha a opção de SAIR.





Perguntas?

