

Construção de Compiladores 1 - 2015.1 - Prof. Daniel Lucrédio

Aula 01 - Introdução - roteiro

Demonstração 1 - Exemplo HTML

1. Criar novo arquivo index.html e abrir no browser

```
<html>
<head>
<title>Exemplo HTML</title>
</head>
<body>
    Olá, <b>pessoal</b>! Eu sou um exemplo de HTML<br/><br/>
    <p>Boa sorte na disciplina de Compiladores!</p>
</body>
</html>
```

2. Abrir algum artigo latex e mostrar rodando

Demonstração 2 – Ferramentas de produtividade

1. Abrir Eclipse/NetBeans
2. Criar um novo projeto Java (DemoCompiladores.Aula01)
3. Criar uma nova classe compiladores.Produtividade
 - 3.1. Criar o void main, mostrar o autocomplete no System.out
 - 3.2. Criar o metodo1, mostrar o autocomplete
 - 3.3. Criar o metodo2, mostrar o autocomplete com a declaração antes
 - 3.4. Criar o metodo3, mostrar o autocomplete “adivinhandando” os parâmetros

```
package compiladores;

public class Produtividade {
    int metodo1() { return 10; }
    float metodo2() { return 10.0f; }
    void metodo3(int a, float b) { }

    public static void main(String args[]) {
        System.out.println("Alo mundo");
        Produtividade p = new Produtividade ();
        p.metodo1();
        int a = p.metodo1();
        float b = p.metodo2();
        p.metodo3(a, b);
    }
}
```

Demonstração 3 – Estrutura de blocos

1. Abrir Eclipse/NetBeans
2. Criar/abrir projeto Java (DemoCompiladores.Aula01)
3. Criar uma nova classe compiladores.EstruturaDeBlocos

```
package compiladores;

public class EstruturaDeBlocos {
    int a = 1; int b = 1;
    void metodo1() {
        int a = 2;
        int b = 2;
        System.out.println(a + " " + b);
    }
    void metodo2() {
        System.out.println(a + " " + b);
    }
    public static void main(String[] args) {
        EstruturaDeBlocos edb = new EstruturaDeBlocos();
        edb.metodo1();
        edb.metodo2();
    }
}
```

Demonstração 4 – Passagem de parâmetros

1. Abrir Eclipse/NetBeans
2. Criar/abrir projeto Java (DemoCompiladores.Aula01)
3. Criar uma nova classe compiladores.PassagemParametros

```
package compiladores;

public class PassagemParametros {
    static int proc(int a, int b) {
        a = a + 5;
        b = b + 5;
        return a + b;
    }

    static String proc2(String a, String b) {
        a += "ster";
        b += "gol";
        return a + b;
    }

    public static void main(String args[]) {
        int valor1 = 10;
        int valor2 = 20;
        int valor3 = proc(valor1, valor2);
        System.out.println("Valor1="+valor1);
        System.out.println("Valor2="+valor2);
    }
}
```

```

        System.out.println("Valor3="+valor3);

        String valor4 = "bla";
        String valor5 = "ble";
        String valor6 = proc2(valor4,valor5);
        System.out.println("Valor4="+valor4);
        System.out.println("Valor5="+valor5);
        System.out.println("Valor6="+valor6);
    }
}

```

3. Mudar proc2 para usar StringBuffer

```

static StringBuffer proc2(StringBuffer a, StringBuffer b) {
    a.append("ster");
    b.append("gol");
    return new StringBuffer(a.toString() + b.toString());
}

...

StringBuffer valor4 = new StringBuffer("bla");
StringBuffer valor5 = new StringBuffer("ble");
StringBuffer valor6 = proc2(valor4,valor5);

```

4. Fazer a chamada para proc2 passando o mesmo objeto (valor4)

```

StringBuffer valor6 = proc2(valor4,valor4);

```