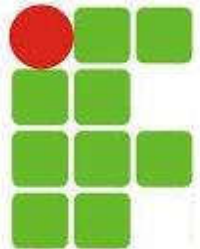


---

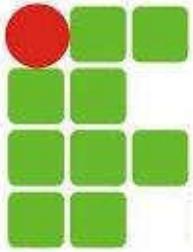
# ALGORITMOS

**Professor: Diego Oliveira**



**Aula 06 -  
Tipos Primitivos de Dados**

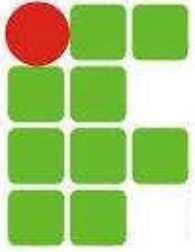




# Netbeans

- Para testar nossos exemplos criaremos uma classe no Netbeans
- Nesta disciplina não será visto nada de orientação a objetos, trabalharemos apenas dentro do método **main** que é criado automaticamente pelo Netbeans no momento da criação do projeto
- Vamos ver como fazer isso nos próximos slides





# Netbeans

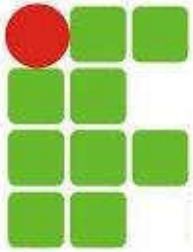
---

- Abra o Netbeans, que estará na área de trabalho do computador do laboratório



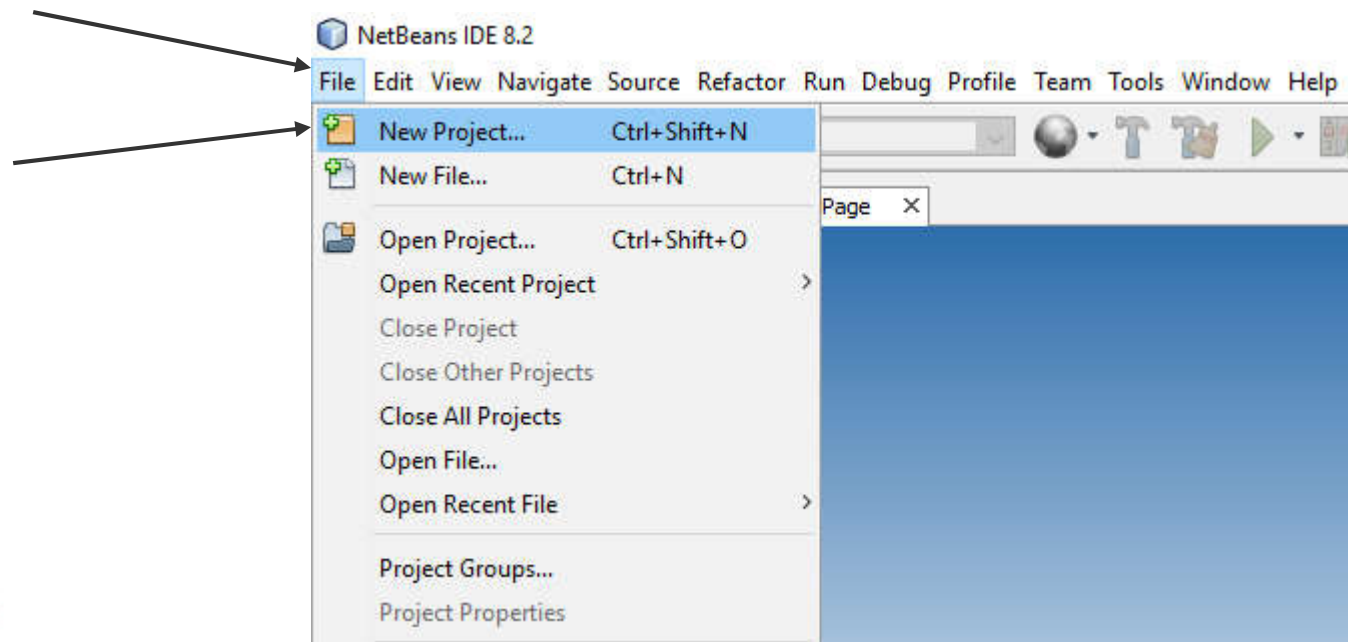
- Aguarde até que a tela principal do programa seja inicializada

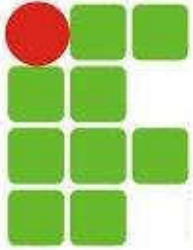




# Netbeans

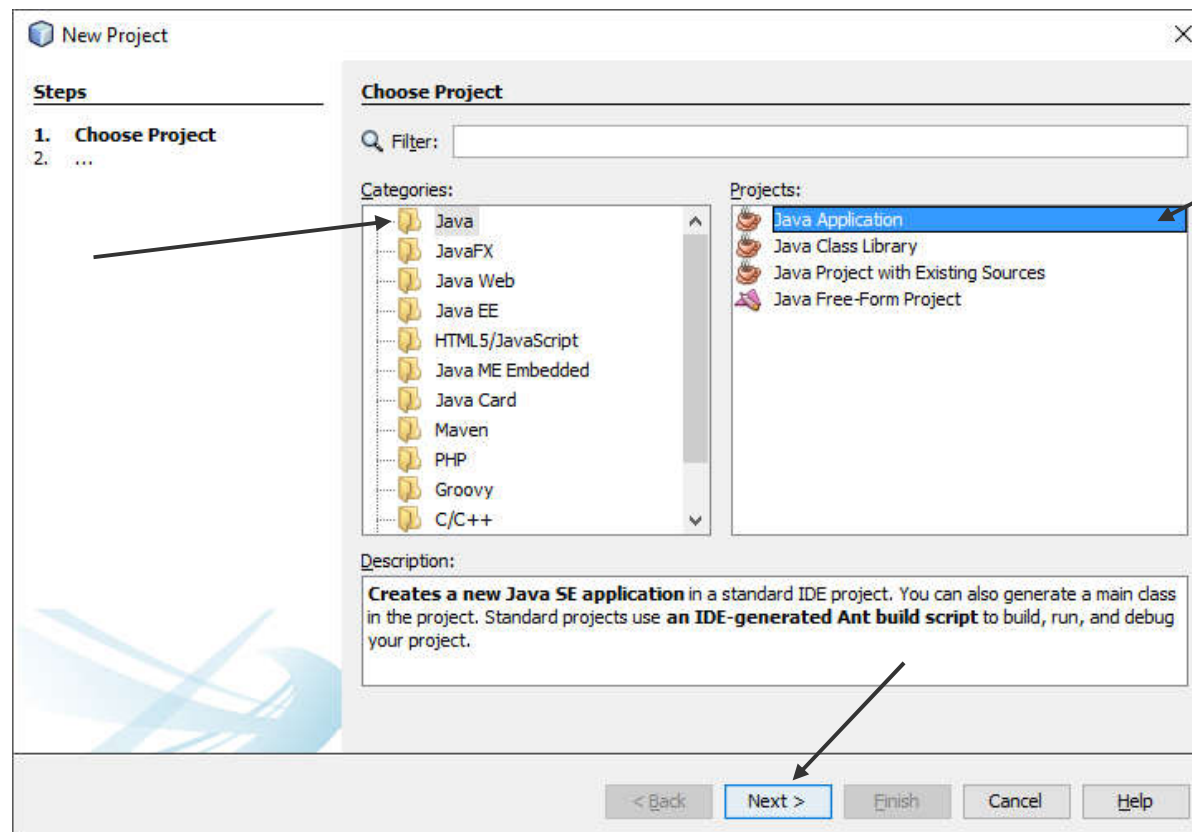
- Depois clique em Arquivo (File) e Novo Projeto (New Project):

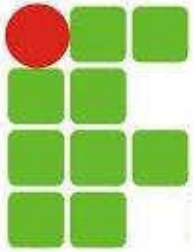




# Netbeans

- Escolha a pasta Java, depois Aplicação Java (Java Application) e clique em Próximo (Next):





# Netbeans

- Escolha um nome para seu projeto e deixe marcado Criar Classe Principal (Create Main Class):

New Java Application

**Steps**

1. Choose Project
2. **Name and Location**

**Name and Location**

Project Name: OlaMundo

Project Location: C:\Users\diego\Documents\NetBeansProjects Browse...

Project Folder: C:\Users\diego\Documents\NetBeansProjects\OlaMundo

Use Dedicated Folder for Storing Libraries

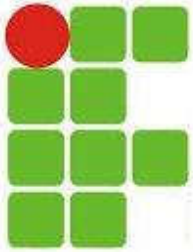
Libraries Folder: Browse...

Different users and projects can share the same compilation libraries (see Help for details).

Create Main Class olamundo.OlaMundo

< Back Next > Finish Cancel Help





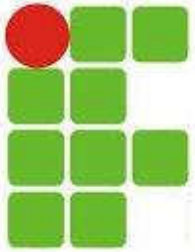
# Netbeans

- Será criado automaticamente um código como o mostrado abaixo:

```
OlaMundo.java x
Source History
1 package olamundo;
2
3 public class OlaMundo {
4     public static void main(String[] args) {
5         // TODO code application logic here
6     }
7 }
```



- Este código ainda não faz nada, vamos editá-lo para que imprima “Olá Mundo!”



# Netbeans

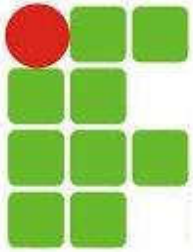
- A linha 5 deve ser alterada e o código ficará assim:

```
1  package olamundo;
2
3  public class OlaMundo {
4      public static void main(String[] args) {
5          System.out.println("Olá Mundo!");
6      }
7  }
```

- Para executar o código aperte F6 e observe a saída na aba Saída (Output) do Netbeans na parte inferior







# Netbeans

- Observe a saída:

The screenshot displays the NetBeans IDE 8.2 interface. The main editor window shows the source code for `OlaMundo.java`:

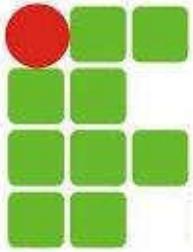
```
1 package olamundo;  
2  
3 public class OlaMundo {  
4     public static void main(String[] args) {  
5         System.out.println("Olá Mundo!");  
6     }  
7 }
```

The bottom panel shows the `Output - OlaMundo (run)` window with the following output:

```
run:  
Olá Mundo!  
BUILD SUCCESSFUL (total time: 0 seconds)
```

An arrow points from the `run:` label in the output window to the `run:` label in the code editor's line numbers.

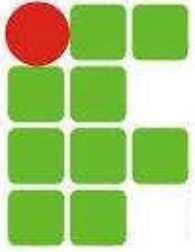




# Netbeans

- O `System.out.println()` é um método que imprime um texto
- Esse texto está entre aspas duplas e aparece na cor laranja
- Também é possível imprimir números e valores de variáveis, como veremos a seguir
- Para usar variáveis precisamos declará-las e para tal devemos escolher um **tipo**, um nome e um valor

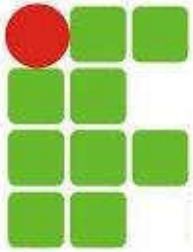




# Tipos Primitivos

- Cada linguagem de programação tem seus tipos primitivos, no Java são 8:
  - boolean
  - char
  - byte
  - short
  - int
  - long
  - float
  - double





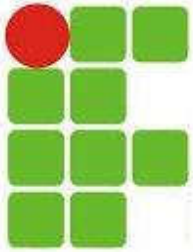
# int

- Nesse caso foi criada uma variável chamada 'idade' com valor '30' do tipo 'inteiro':

```
public class OlaMundo {  
    public static void main(String[] args) {  
        int idade = 30;  
        System.out.println("A idade é " + idade);  
    }  
}
```

- Ao executar este código a saída será 'A idade é 30'
- Podemos declarar valores reais, com casas decimais utilizando os tipos float ou double





# double

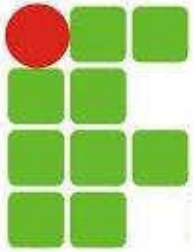
- Vamos criar uma variável que representa um salário com seus respectivos centavos

```
public class OlaMundo {  
    public static void main(String[] args) {  
        double salario = 954.53;  
        System.out.println("A idade é " + salario);  
    }  
}
```

- Lembrando que em inglês a separação de casas decimais é representada pelo PONTO e não pela vírgula, como no português



- E o verdadeiro e falso professor???



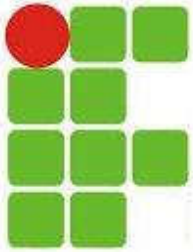
# boolean

- Neste caso foi criada uma variável boolean com valor verdadeiro (true)

```
public class OlaMundo {  
    public static void main(String[] args) {  
        boolean isProfessor = true;  
        System.out.println("Diego é professor? " + isProfessor);  
    }  
}
```

- Java utiliza CamelCase para representar variáveis com mais de uma palavra, a segunda fica com a inicial maiúscula: meuNome, minhaldade, salarioMinimo...





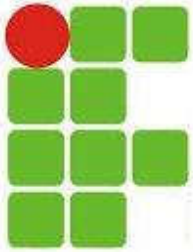
# char

- Para representar um caractere utilizamos o tipo 'char':

```
public class OlaMundo {  
    public static void main(String[] args) {  
        char genero = 'm';  
        System.out.println("O gênero é " + genero);  
    }  
}
```

- Não se coloca acento em nome de variável, o Java permite, porém não é uma boa prática de programação!





# String

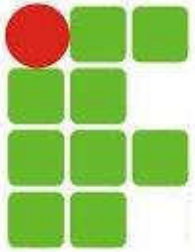
- E para salvar um texto? Neste caso o Java utiliza a classe String, que é um conjunto de caracteres:

```
public class OlaMundo {  
    public static void main(String[] args) {  
        String meuNome = "Diego Oliveira";  
        System.out.println("O nome do professor é " + meuNome);  
    }  
}
```

- O tipo primitivo 'char' só permite um caractere enquanto a String permite um texto completo, porém **não é um tipo primitivo!**



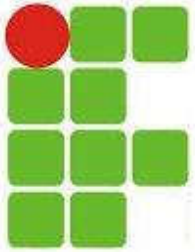




# Exemplo com vários tipos

- Podemos criar diversas variáveis e imprimir seus valores todos de uma vez:

```
public class OlaMundo {  
    public static void main(String[] args) {  
        String meuNome = "Diego Oliveira";  
        int minhaIdade = 30;  
        double meuPeso = 77.5;  
        boolean isLegal = true;  
        char meuGenero = 'm';  
        System.out.println("O nome do professor é " + meuNome);  
        System.out.println("A idade do professor é " + minhaIdade);  
        System.out.println("O peso do professor é " + meuPeso);  
        System.out.println("O professor é legal? " + isLegal);  
        System.out.println("O genero do professor é " + meuGenero);  
    }  
}
```



# Exemplo com operações

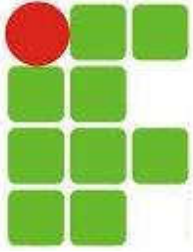
- Também podemos realizar operações aritméticas com os valores das variáveis:

```
public class OlaMundo {  
    public static void main(String[] args) {  
        double salario = 1200.00;  
        System.out.println("Em um ano esta pessoa ganha " + salario * 12);  
        System.out.println("Metade do salário desta pessoa é " + salario / 2);  
    }  
}
```

- Neste caso será impresso o valor ganho em um ano, ou seja o salário vezes 12 e a metade do salário



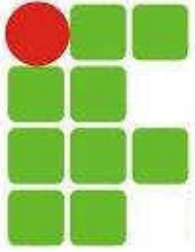
- As operações de + e - também são possíveis



# Exercício

- Crie um projeto com o nome MeusDados e salve na sua pasta do IFRN
- Edite o código dentro do método MAIN para que ele imprima 10 informações sobre você
- Utilize 10 variáveis de pelo menos 5 tipos diferentes, como mostrado na aula
- O texto impresso deve ser claro!
- Imprima a quantidade de dias, horas, minutos e segundos de sua vida utilizando a variável idade como base





# Perguntas?

