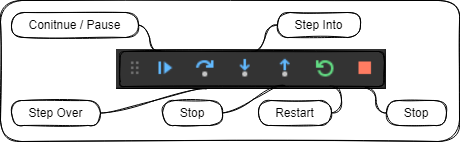
**Introdução à Programação - Unidade 3**

* Boas práticas de programação -> legibilidade do código
  + [Indentação de código](https://pt.wikipedia.org/wiki/Indenta%C3%A7%C3%A3o);
  + VSCode Formatar Documento (menu de contexto no código ou teclas ⇧⌥F macOs/Linux e teclas SHIFT+ALT+F).
*  Criar um novo projeto no VSCode para representar a Unidade 3 com o nome "Unidade3". Assim, lembrar como criar projeto e classe.
*  Resgatar conteúdo da Unidade 3 e mostrar o que é classe e o que é o método main de forma superficial. Lembrar que todo o código vai dentro do método main!
*  Resolver os primeiros exercícios (exemplos) [Atividades - Aula](https://github.com/lucianapda/disciplinaIPSegunda/blob/main/Unidade3/atividadeAula.md).

Desenvolvimento de código (mostrar um exemplo de uso do DEBUG)  
Executar código  
RUN  
DEBUG -> Testes de código



BreakPoint: marcar ponto de "parada" no código (círculo vermelho)

Controle da execução:

Stop: quadrado vermelho

Continue: continuar e para no próximo BreakPoint

Step Over: executar linha apor linha (instrução)

### [Atividades - URI](https://github.com/lucianapda/disciplinaIPSegunda/blob/main/Unidade3/atividadeUri.md)

* Apresentar o URI, mostrar como eles fazem para entrar na turma.
* Lembrar que devem utilizar o **Java 14 no URI**.
* Resolver os dois primeiros exercícios do URI no VSCode e depois colocar no URI.
* Para o Uni3Uri1002.java será necessário usar formatação, [DecimalFormat](https://github.com/lucianapda/disciplinaIPSegunda/blob/main/Exemplos/src/ExemploDecimalFormat.java).

#### Rabiscos do processo de entender os enunciados

*  Classe **Math** **não** precisam importar ela (pacote base java.lang).
*  Mostrar a documentação do OpenJDK os métodos da classe Math (<https://devdocs.io/openjdk~17/java.base/java/lang/math>) e, se der tempo, de outras classes como Scanner para acostuma-los a utilizar a documentação.
*  Classe [Scanner](<https://devdocs.io/openjdk~17/java.base/java/util/scanner> "Scanner).