

# SISB020 - Desenvolvimento de Software para Dispositivos Móveis

## Aula 5: Parte 2

Prof. Thiago Cavalcante


# Observação sobre a avaliação



# Flutter

Página: <https://flutter.dev>

# Principais características

- Uma base de código, uma linguagem ( Dart)
- Performance nativa (compilação para código de máquina)
- Apps para celular, web, desktop (em fase de desenvolvimento)
- UI flexível, bonita, com experiência nativa
- Desenvolvimento rápido com a função ***hot reload***
- Código aberto

# Instalação do Flutter

- Pré-requisitos: SO, espaço em disco, ferramentas
- Instalar o SDK do Flutter
- Atualizar a variável de ambiente `PATH`
- Rodar `flutter doctor`
- Para desenvolvimento Android: Instalar Android Studio (vem com o Android SDK) e Emulador
- Instalar um **editor de texto/IDE**



# Instalação do Flutter: Windows

- ▶ Parte 1: Instalando o SDK do Flutter
- ▶ Parte 2: Instalando o Android Studio
- ▶ Parte 3: Instalando o Emulador de Android

# Editores de texto recomendados


- Android Studio (*+ Plugins Flutter e Dart*)
- Visual Studio Code (*+ Extensões Flutter e Dart*)
- Emacs (*+ Pacote lsp-dart*)

# Editores de texto recomendados


-  Android Studio (*+ Plugins Flutter e Dart*)
-  Visual Studio Code (*+ Extensões Flutter e Dart*)
  - Configurações recomendadas
- Emacs (*+ Pacote lsp-dart*)



# Alternativa: **Repl.it**

- Usado para rodar Dart no navegador
- Não requer nenhuma instalação
- Funciona no celular 
- Extra: serve para outras linguagens

# Alternativa: Dartpad

- Usado para rodar Dart e **Flutter (!)** no navegador
- É possível ver as interfaces do Flutter
- Não requer nenhuma instalação
- Funciona no celular  (interface não fica muito boa)



# Dart

Página: <https://dart.dev>

# Principais características

- Otimizado para UI
  - Programação assíncrona
  - Operadores e funções especiais voltados para UI
  - Sintaxe familiar (Java, C++, JavaScript)
- Desenvolvimento produtivo
  - ***Hot reload***
  - Ferramentas de análise, log, debug etc.
- Rápido em várias plataformas
  - Compilação para arquiteturas ARM (celulares) e x86\_64 (computadores)
  - Compilação para JavaScript (web)
  - Pode ser usado para desenvolvimento de *backend*

# Primeiro programa: Olá, mundo!

```
// Início da execução do programa  
void main() {  
    print('Olá, mundo!');  
}
```

# Conceitos importantes

- Tudo que você armazena em uma variável é um objeto. Todos os objetos herdam da classe `Object` (números, funções, `null`)
- Dart é uma linguagem *fortemente tipada*, mas o programa pode "adivinhar" o tipo de uma variável. O tipo especial `dynamic` aceita qualquer valor
- Identificadores podem começar com uma letra ou `_`, seguido de uma combinação desses mesmos caracteres e números
- Dart possui *expressões* (com valor) e *instruções* (sem valor)
- Dois tipos de problemas: alertas ⚠ e erros 🚫. Erros podem ser de execução ou compilação