
LITERACIA3.0

ENTENDENDO, PRODUZINDO E INOVANDO COM TECNOLOGIA



AULA 05 –INTRODUÇÃO A PROGRAMAÇÃO

1º Edição

INTRODUÇÃO A PROGRAMAÇÃO

O que é programação?

Embora a tecnologia esteja tão avançada, os computadores ainda não 'pensam' exatamente, no sentido literal da palavra. Eles não tem sentimento e nem raciocinam! Precisamos ajudá-los, afinal eles simplesmente executam o que lhe são ordenado.

As pessoas simplesmente não percebem, ou não precisam, ou acham que não precisam, e acabam por se contentar com o Facebook por exemplo, mas você pode sim ter o total controle sobre uma máquina dessas e usá-la para seus propósitos, quaisquer que sejam os seus planos, saber programar vai te ajudar. Em qualquer profissão por exemplo, você pode ensinar o computador a fazer muitas coisas e tornar automático muitos processos!

Mas e aí?! O que é programação?! Como começar a programar?! Pra que serve e onde é usado programação de computadores?! Com programação podemos:

- criar jogos;
- criar sistemas operacionais, tipo Windows e Android;
- criar programas para fazer suas tarefas no dia-a-dia para facilitar seu trabalho, como o Calendário do seu celular;
- fazer pesquisas científicas e/ou acadêmicas;
- realizar cálculos qualquer tipo;
- criar aplicativos, como Snapchat;
- criar o próximo Facebook;
- milhões, mas milhões de outras utilidades mesmo;

Programar é você dizer para o computador o que você quer, e ele simplesmente fará.

Mas como dizer isso? Falando? Em português? Inglês? E se eu gritar? Ou chutar a máquina? Resolve?

Não!

O que fazer então? Como me comunicar com ele? Através de linguagens de programação, ué! Vamos falar logo mais sobre os tipos de linguagem de programação.

Conceitos Básicos

VARIÁVEL

Vamos começar com um exemplo! Que tal darmos um nome para o seu ano de nascimento? Em outras palavras, vamos atribuir um nome para ele, por exemplo nascimento. Fazemos isso da seguinte forma:

```
>>> nascimento = 2003
```

Primeiro escrevemos o nome da **variável** do lado esquerdo e dizemos que isso é igual ao valor que colocamos do lado direito.

A partir desse momento, sempre que nos referirmos ao nome "nascimento", o computador saberá que ele se refere ao valor que atribuímos acima.

Agora, vamos fazer as contas da idade que você terá em 2020, 2050 e 2075, só que vamos usar essa variável que acabamos de criar.

```
>>> 2020 - nascimento
```

```
17
```

```
>>> 2050 - nascimento
```

```
47
```

```
>>> 2075 - nascimento
```

```
72
```

Observe que eu usei o nome variável para me referenciar à nascimento, isso porque esse valor também pode mudar, se quisermos. Podemos imaginar

uma variável como um pedacinho da memória do computador que armazena um determinado valor e pode ser chamado por um nome que escolhemos. **Imagina que é uma caixinha que você pode colocar coisas dentro e você coloca um rótulo, pra quando você for procurar pelo que está lá dentro você ache mais fácil.** Deu pra entender?!

PROCEDIMENTO

A partir de agora quando alguém falar de Procedimento, Função e Algoritmo quero que pensem em um passo a passo! Eles tem significados específicos, mas nesse momento não precisamos saber disso! :)

Lembra quando sua mãe pergunta se você está ponto pra ir pra escola? Ela não precisa falar todas as vezes todos os passos e “coisas” que você tem que ver para poder responder que sim, afinal vocês dois já sabem qual é o passo a passo explícito para ir pra escola. Não percebemos isso porque nosso cérebro é muito rápido e já estamos “no automático”.

Então procedimento para um computador é uma sequência de passos que recebe um nome, ok!? Aí só de falar esse nome já sabemos todos os passos! :)

Tipos de Linguagem

PROCEDURAL

É um dos mais conhecidos! Ela executa tarefas em série, ou seja, uma após outra, manipulando variáveis baseando-se em ordens ou comandos, onde variáveis são lidas num processo de entrada, manipuladas ou alteradas e por fim atualizando variáveis e gerando a saída.

Exemplos de linguagens desse tipo: Ada, Algol, Basic, C, Cobol, Fortran e Pascal.

FUNCIONAL

Essencialmente desenvolve programas escritos em linguagem definida por expressões e declarações, considerando a computação como uma avaliação de funções matemáticas. Oferece-nos também a possibilidade de programar onde podemos determinar o que se pretende construir e não como será construído. Não utiliza o conceito de atribuição pelo fato de que

os programas são formados por definições de funções. Valores podem ser criados, mas não podem ser modificados.

Exemplos de linguagens desse tipo: LISP, Scheme, Miranda e Haskell.

ORIENTADA A OBJETOS

Aqui a programação é fundamentada na composição e interação de várias unidades de softwares denominados objetos os quais se relacionam e trocam mensagens entre si. Estes objetos são chamados de classes.

Nestas classes cada comportamento é denominado como método (ações). Já os estados de uma classe são chamados de atributo. Em cada método e atributo podemos também definir a forma de relacionamento com os outros demais objetos.

Exemplos de linguagens desse tipo: Smalltalk, Python, Ruby, .Net, C#, C++ e Java.

ORIENTADA A EVENTOS

É quando você escreve código para responder a eventos. Na programação orientada a eventos, uma rotina especializada em monitorar os eventos avisa o código especializado em responder a um determinado evento que aquele evento que ele esperava ocorreu; e então o código recém avisado responde ao evento.

Exemplos de linguagens desse tipo: Visual Basic, Delphi, Javascript e Blockly (é essa que vamos trabalhar daqui pra frente!)

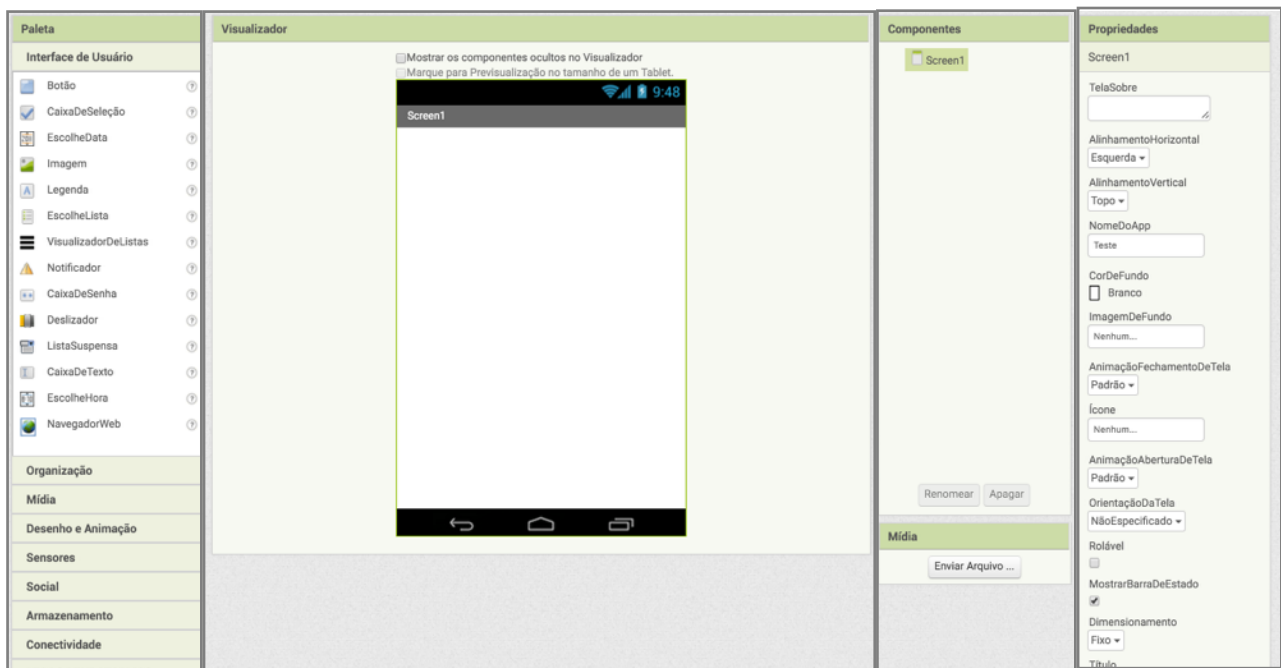
O que é o AppInventor?

Bele! Agora que já sabemos de tudo isso vamos começar a programar? Aqui vamos usar uma ferramenta chamada AppInventor, tudo começou no Google e hoje a ferramenta é melhorada pelo MIT. Vamos entrar um pouquinho mais no detalhe da ferramenta agora :)

O conjunto é composto por duas seções: **o App Inventor Designer e o App Invento Blocos**, cada uma com uma função específica.

App Inventor Designer

O Designer é a tela inicial de um projeto. É aqui que você desenha seu aplicativo, escolhendo a posição dos botões e imagens, inserindo fotos, droplists, checkboxes e outros componentes disponíveis para a construção de um programa. Ele é dividido em quatro colunas.



1. PALETA

2. VISUALIZADOR

3. COMPONENTES 4. PROPRIEDADES

1. Paleta

A primeira coluna é onde ficam todos os componentes utilizáveis num aplicativo. Esta paleta é dividida em seções para facilitar a localização dos componentes, que vão dos básicos (botões, imagens e textos) até uma seção exclusiva para integração com ferramentas de Lego Mindstorms. Para utilizar um desses componentes basta clicar sobre ele e arrastar para cima da segunda coluna, chamada Visualizador.

2. Visualizador

Na coluna central, o usuário pode organizar cada um de seus objetos, montando o aplicativo como ele deve ser. Uma janela de exibição simula a tela de um smartphone com o sistema operacional Android, apresentando

uma versão próxima da final ao programador, à medida que ele organiza o espaço de uso do programa. Todos os itens adicionados da Paleta ao Visualizador são apresentados na terceira coluna, chamada de Componentes.

3. Componentes

Na coluna de componentes, ficam armazenados todos os itens adicionados, sejam eles visíveis ou não na tela do programa. Dessa forma, fica muito mais simples selecionar cada objeto, pois eles estão listados de forma ordenada e acessível. Aqui também é possível renomear cada item. Assim, você pode chamar os componentes por nomes que façam sentido para o seu projeto. É muito mais fácil encontrar cada coisa se você mesmo criar um nome específico para ela, Como “botão de som” em vez de “Button1”. É possível também inserir arquivos de mídia pela terceira coluna. Clicando no botão Enviar Arquivo você importa sons, fotos e vídeos do seu computador para o servidor do App Inventor, e eles ficam disponíveis para que você possa usar no projeto.

Clicar sobre qualquer um dos itens da lista na coluna Componentes permite que você possa editar seus detalhes na quarta coluna, chamada de Propriedades.

4. Propriedades

Esta é a coluna mais importante do Designer, já que aqui você pode definir os tamanhos e conteúdos dos textos de botões e caixas de informação, tamanho das imagens, cores de fundo, largura e altura de objetos, enfim, você consegue deixar as coisas do jeito que você quiser. Essas e muitas outras configurações são aplicadas instantaneamente na tela do Visualizador.

Uma vez que seu aplicativo já tem a carinha das primeiras telas, é hora de começar a atribuir funções a cada um dos componentes que você selecionou. Para isso é preciso clicar no botão “Blocos” no canto superior direito.

App Inventor Blocos

O Editor de Blocos permite programar de forma simples.

Após instalar Java no seu computador (por isso não conseguimos usar o celular ou o tablet para programar com App Inventor), você pode começar a fazer a mágica da programação. Acessando o “Blocos” você pode começar a associar ações para cada item do seu programa. Usando uma interface simples e intuitiva, a construção do aplicativo parece muito com montar um quebra-cabeça.

O menu na lateral esquerda fornece duas abas de comandos: Internos e Blocos dos componentes daquela janela. Todos os objetos que você inseriu em seu programa terão comandos de início na aba que tem o nome da sua tela, ao passo que os comandos de execução estão localizados na aba Internos. A combinação de um ou mais comando dos seus blocos e comandos internos forma uma ação completa. E para facilitar a construção das ações, os comandos são estruturados como peças de quebra-cabeças. Apenas funções compatíveis se encaixam.

Por exemplo, caso você crie um botão em seu programa que deve tocar um som toda vez que for tocado, basta arrastar o comando do evento/início (dentro do submenu do botão que você deseja clicar) “quando NomeDoBotao.Clique fazer” para o campo à direita. Após isso, na mesma guia, selecione o componente de som e arraste a opção “chamar.NomeDoSom.Iniciar” para dentro do outro componente, criando uma ação completa. Quando o botão for clicado no seu programa, o som vai tocar.

Instalação do app em um Android

O jeito mais fácil de instalar o seu aplicativo em um dispositivo Android é sem dúvida através de um leitor de QR Code, sabe aquele quadradinho cheio de outros quadradinhos?!

O App Inventor te dá a opção de gerar um desses para acessar o seu aplicativo e baixar no seu celular. Para isso basta baixar um leitor de QRcode no seu celular, se você digitar isso na busca no Google Play você pode baixar o primeiro que aparecer :)

Aí é só clicar no menu lá em cima em Compilar e depois selecionar a primeira opção: App (fornecer o QR code para o .apk) aí é só seguir o passo a passo que o próprio celular vai te guiando :)

Meu primeiro aplicativo

Tutorial da bola de cristal.

Veja os seguintes vídeos:

INTRO: <https://www.youtube.com/watch?v=mDzMkMn4eM0&index=10&list=PLe9iGEQ2t4s6aHM1jUt-F44laAb9tichR>

Bola de Cristal - PARTE 1 [Designer]: <https://www.youtube.com/watch?v=CPCIA20Csec&index=11&list=PLe9iGEQ2t4s6aHM1jUt-F44laAb9tichR>

Bola de Cristal - PARTE 2 [Blocos]: <https://www.youtube.com/watch?v=JwAWtqt24-U&index=12&list=PLe9iGEQ2t4s6aHM1jUt-F44laAb9tichR>

Bola de Cristal - Parte 3 [Instalação]: <https://www.youtube.com/watch?v=Xx80l3Crleg&index=13&list=PLe9iGEQ2t4s6aHM1jUt-F44laAb9tichR>

Cenário offline: a alternativa aqui é trabalhar com o AppInventor offline. A preparação do ambiente pode ser bastante trabalhosa, mas todas as informações se encontram aqui: <https://sourceforge.net/projects/allivecomplete>.

Outra possibilidade é trabalhar com a seguinte atividade de programação offline: <http://csunplugged.org/wp-content/uploads/2014/12/portuguese-brazil-12.pdf>

Por fim, a versão offline do Scratch pode ser utilizada. Mais informações encontradas aqui: <https://scratch.mit.edu/scratch2download/> (é possível trocar o idioma para português no rodapé da página). Para a atividade utilizar o Material de Apoio que está na mesma página.