

Introdução e história da linguagem Java

Disciplina: Desenvolvimento de Aplicações



Conteúdos:

Introdução e história da linguagem Java.

Bloco 1

Qual é o seu humor hoje?

Qual é a escultura que mais lhe representa?



1



2



3



4



5



6

Tente responder

Você já se perguntou como as linguagens de programação moldam o mundo digital em que vivemos?

??



Linguagens de programação

As linguagens de programação são a base de praticamente todas as tecnologias que usamos diariamente, desde os aplicativos móveis até os sistemas complexos de negócios. Como base da tecnologia moderna, elas permitem que os humanos se comuniquem com computadores e criem *softwares* funcionais. Nesse contexto, existem diversas linguagens, cada uma delas com as suas características e finalidades específicas.

Java nos dias atuais

Java, uma das linguagens mais populares, desempenha um papel crucial em várias áreas da tecnologia, como desenvolvimento *web*, *mobile* e sistemas empresariais. Atualmente, ela apresenta dois pontos importantes.

Relevância

Java é uma das linguagens mais relevantes e amplamente usadas na indústria de *software*.

Aplicações

Pode ser aplicada em desenvolvimento *web*, *mobile*, sistemas empresariais, Internet das Coisas (IoT), inteligência artificial, entre outros.

Popularização da linguagem Java



História

Lançada pela *Sun Microsystems* em 1995, rapidamente se tornou popular.



Versatilidade

Sua versatilidade a levou a ser adotado por diversas áreas da tecnologia.



Sucesso

Ganhou popularidade devido à sua flexibilidade e capacidade de ser usada em diferentes plataformas, o que a tornou uma escolha ideal para diversas aplicações.

Diferentes áreas de atuação

A linguagem Java é amplamente utilizada no desenvolvimento de aplicativos *web* e *sites* dinâmicos, impulsionando a interação e a experiência do usuário.

Desenvolvimento *web*

Plataformas como *Spring* e Java EE são fundamentais para construir aplicações *web* robustas e escaláveis.

Desenvolvimento *mobile*

Java é a base para o desenvolvimento de aplicativos Android, dominando o mercado *mobile*.

Sistemas empresariais

Muitas corporações utilizam Java para criar sistemas empresariais complexos e seguros.

Filosofia da linguagem Java



Portabilidade

O código Java pode ser executado em diferentes plataformas sem modificações.



Legibilidade

Prioriza a legibilidade e simplicidade do código, facilitando a manutenção.



Ampla comunidade

Grande comunidade de desenvolvedores, fóruns e bibliotecas disponíveis.

Impactos da linguagem Java

- Uma aplicação Java criada em um sistema operacional pode ser executada em Windows, Linux e macOS sem grandes modificações, isso reduz os custos de desenvolvimento e manutenção, tornando o *software* mais acessível;
- A linguagem Java revolucionou a forma como o *software* é desenvolvido, proporcionando soluções mais eficientes e robustas. Além de ter influenciado no desenvolvimento de outras linguagens e tecnologias.

Vamos praticar?

Formem duplas para realizar a atividade.

Primeiro momento

10 min

- Pesquisem no celular exemplos de empresas que utilizam Java;
- Anotem no caderno o que acharem interessante.

Segundo momento

15 min

Compartilhem as descobertas com a turma e debatam sobre as linguagens de programação.



Bloco 2

Duas verdades e uma mentira



- Formem duplas;
- Em cinco minutos, escrevam duas informações verdadeiras e uma falsa sobre a linguagem Java;
- O(A) professor(a) vai escolher algumas duplas para ler as frases criadas e a turma deve tentar acertar qual é a mentira.

Programação orientada a objetos

A Programação Orientada a Objetos (POO) é um paradigma de programação que organiza o código em torno de objetos, permitindo modelar o mundo real de maneira mais fiel. É uma abordagem utilizada para formular programas de *software* baseados em objetos.



Dê um *play* no conhecimento!



O que é Programação Orientada a Objetos (POO)?



Classes e objetos

Classes

São estruturas que definem os atributos e comportamentos que os objetos de um tipo específico terão.

Objetos

São instâncias das classes, representando entidades do mundo real.



Está certo?

Na classe carros, podemos ter objetos como bicicleta e motocicleta, pois também são meios de transporte.

A afirmativa é...?

a

Verdadeira.

b

Falsa.



Está certo?

Na classe carros, podemos ter objetos como bicicleta e motocicleta, pois também são meios de transporte.

A afirmativa é...?

a

Verdadeira.

b

Falsa.



Alguns conceitos



Herança

A herança permite criar classes baseadas em outras classes, compartilhando atributos e comportamentos. Isso promove a reutilização de código e uma estrutura hierárquica.



Encapsulamento

O encapsulamento permite ocultar detalhes internos de uma classe e fornecer uma interface controlada para interagir com ela. Isso melhora a segurança e a manutenção.



Polimorfismo

O polimorfismo permite que objetos de diferentes classes sejam tratados de maneira uniforme, utilizando um único nome para várias formas.

Reutilização de código e organização modular

A POO promove a reutilização de código através de herança e composição, facilitando a manutenção e expansão do *software*. A organização modular melhora a legibilidade e a manutenção do código.

Benefícios da POO em Java

A POO promove organização, reutilização de código, modularidade e facilita a representação de conceitos do mundo real.

Reutilização de código

Permite criar classes e objetos genéricos que podem ser reutilizados em diferentes contextos.

Organização modular

Facilita a organização do código em partes menores e mais gerenciáveis.

Manutenção simplificada

Mudanças em uma parte do código afetam apenas a parte relevante, sem impactos em outras áreas.

Vamos praticar?

Formem cinco grupos para realizar a atividade.

Primeiro momento

15 min

Criem um *folder* explicando o que é POO e exemplificando classe, objeto, herança, encapsulamento e polimorfismo. Escrevam em uma linguagem clara.

Segundo momento

10 min

Apresentem para a turma o que foi desenvolvido.



Bloco 3

Independência de plataforma da linguagem Java

A linguagem Java é conhecida por sua independência de plataforma, que permite que programas escritos em Java sejam executados em diferentes sistemas operacionais sem a necessidade de recompilação. Isso acontece porque o código Java é compilado para um formato intermediário (*bytecode*), que é interpretado pela Máquina Virtual Java (JVM) específica de cada plataforma.

Máquina Virtual Java (JVM)

A JVM interpreta o *bytecode* gerado pelo compilador Java, tornando os programas independentes de plataforma.

Flexibilidade

O mesmo código pode ser executado em Windows, macOS e Linux.

Redução de custos

Desenvolvedores não precisam criar versões diferentes para cada sistema.

Alcance maior

Aplicativos Java podem ser distribuídos amplamente, atingindo diversos usuários.

Coleta de lixo automática (*Garbage Collection*)

Definição

A coleta de lixo automática é um recurso da linguagem Java que gerencia automaticamente a alocação e liberação de memória para objetos.

Problema

Sem gerenciamento adequado, programas podem ficar sobrecarregados com objetos não utilizados, resultando em vazamento de memória.

Funcionamento da coleta de lixo automática

A JVM monitora a alocação de memória e identifica objetos que não possuem mais referências.

O coletor de lixo remove esses objetos, liberando espaço para novos objetos.

Diferentes algoritmos de coleta de lixo são usados para determinar quais objetos podem ser removidos.

Tente responder

Entendendo os conceitos, quais são os benefícios dessa coleta de lixo automática? Cite exemplos que você consegue lembrar de já ter observado.

??





Vantagens



Eliminação de vazamentos de memória

Garante que a memória seja liberada de forma eficiente, evitando que os objetos não utilizados acumulem.



Simplificação do desenvolvimento

Desenvolvedores não precisam se preocupar em liberar manualmente a memória alocada.



Redução de bugs

Menos propenso a erros causados por gerenciamento manual de memória.

Impacto no desempenho

Eficiência: a coleta de lixo pode causar breves pausas no programa enquanto libera a memória não utilizada.

Otimização: as JVMs modernas otimizam a coleta de lixo para minimizar o impacto no desempenho.

Vamos praticar?

Formem duplas para realizar a atividade.

Primeiro momento

15 min

Pesquisem, no celular, um aplicativo que utilize a ferramenta de coleta de lixo automática. Depois, analisem como é feito o processo e quais são os benefícios do aplicativo.

Segundo momento

10 min

Compartilhe com a turma as descobertas e debatam sobre elas.



Bloco 4

Desenvolvimento de aplicativos móveis

A linguagem Java oferece um mundo de oportunidades de desenvolvimento em diversas áreas, graças à sua versatilidade, vantagens e capacidade de adaptação a diferentes cenários tecnológicos.

Aplicações da linguagem Java em diferentes setores

Desenvolvimento de aplicativos móveis



Android Studio e Java são usados para criar aplicativos móveis populares.

Sua robusta biblioteca Android permite a criação de aplicativos para *smartphones* e *tablets*, alcançando um vasto público.

Sistemas bancários



A linguagem Java é utilizada em sistemas de gerenciamento financeiro e transações bancárias seguras.

É uma escolha confiável para sistemas bancários e financeiros devido à sua segurança e desempenho.

Jogos



Jogos em Java são comuns devido à sua capacidade de produzir gráficos e interatividade.

Seu desempenho e recursos gráficos tornam possível a criação de experiências de jogo envolventes.

Aplicações de grande escala



Empresas como Google, Amazon e eBay usam Java para sistemas robustos.

Ótima opção para desenvolver sistemas complexos, como os de gerenciamento empresarial (ERP) e os de planejamento de recursos (ERP), devido à sua capacidade de lidar com aplicações de grande escala.

Vantagens

Segurança

Java inclui recursos de segurança, como verificação de *bytecode*, que reduzem riscos de vulnerabilidades.

Desempenho

A JVM optimiza o código em tempo de execução, resultando em um bom desempenho.

Ecossistema de bibliotecas

Uma vasta coleção de bibliotecas e frameworks acelera o desenvolvimento.

Portabilidade

Graças à JVM, programas Java podem ser executados em diferentes sistemas operacionais sem recompilação.

Máquina Virtual Java (JVM)

A JVM possibilita a portabilidade dos programas Java, permitindo que eles sejam executados em diferentes sistemas operacionais sem a necessidade de recompilação. Sua arquitetura é composta por:

Flexibilidade

Atualizações na JVM podem melhorar o desempenho sem afetar o código-fonte.

Funcionamento

O código-fonte é compilado para *bytecode*, que é interpretado pela JVM.

Portabilidade

A JVM permite a execução do mesmo *bytecode* em diferentes plataformas.

Exemplo de portabilidade

Cenário: um aplicativo Java é desenvolvido e testado em um sistema Windows.

Portabilidade: o mesmo aplicativo pode ser executado em sistemas macOS ou Linux sem alterações.

Economia de tempo e recursos: desenvolvedores não precisam criar versões separadas para cada sistema operacional.

Tente responder

Pense e compartilhe outros exemplos de cenários semelhantes ao anterior.

??



Vantagens para empresas

A otimização da JVM e a compilação *just-in-time* (JIT) permitem que os programas Java alcancem um desempenho competitivo, mesmo em aplicações exigentes.

Economia de custo

O uso de uma única linguagem para múltiplos sistemas reduz despesas.

Manutenção simplificada

Atualizações podem ser feitas globalmente, evitando discrepâncias.

Aumento de produtividade

Um ecossistema rico em bibliotecas e *frameworks* acelera o desenvolvimento.

Vamos praticar?

Formem grupos para realizar a atividade.

Primeiro momento

15 min

Escolham uma aplicação que utiliza Java e apresentem um breve estudo de caso sobre como a linguagem foi aplicada nesse contexto.

Segundo momento

10 min

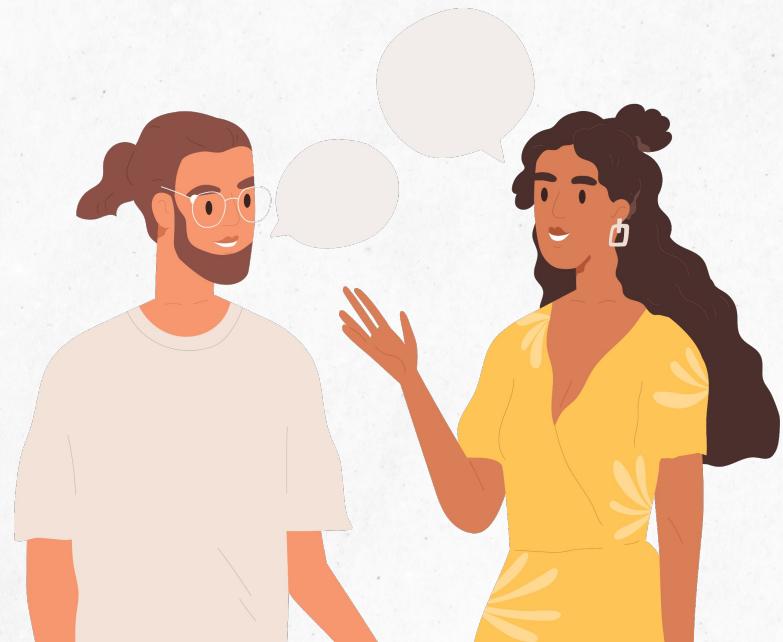
Em grupo, apresentem os prós e contras do uso de Java em diferentes tipos de projetos.



Bloco 5

Fofoca do bem

Que tal contar para a turma, em forma de fofoca, um pouco sobre as utilizações e benefícios da linguagem JAVA?



Vantagens

Declaração de variáveis



Espaço para armazenar dados. Exemplo: 'int idade = 25;'.

Estruturas de controle



If-else, switch (condicionais) e for, while, do-while (loops).

Funções



Conjunto de instruções que realizam uma tarefa.

Declaração de variáveis

Definição

Na linguagem Java, variáveis são utilizadas para armazenar dados, e elas precisam ser declaradas com um tipo específico, como `int`, `double` ou `string`.

Isso ajuda a definir como os dados serão interpretados e manipulados.

java

```
public void saudacao(String nome) {  
    System.out.println("Olá, " + nome + "!");  
}
```



Applets e servlets - conceitos gerais

Applets

Applets são pequenos programas Java que podem ser incorporados em páginas web.

Eles são executados no navegador do usuário e podem interagir diretamente com a interface do dele.

Servlets

Servlets são componentes Java que processam as requisições dos clientes no servidor. Eles podem gerar páginas dinamicamente e interagir com bancos de dados, trazendo conteúdo personalizado para os usuários.

Applets

Exemplo

Applets são incorporados em páginas *web* usando a tag `<applet>`. O navegador exibe o *applet* e permite a interação com o usuário.

html

 Copy code

```
<applet code="ExemploApplet.class" width="300" height="200"></applet>
```

Servlets

Exemplo

Servlets são implantados em um servidor web (por exemplo, Apache Tomcat). O navegador envia uma solicitação para o *Servlet* através de uma URL. Ele processa a solicitação e envia uma resposta ao navegador.

```
java
Copy code

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class ExemploServlet extends HttpServlet {
    public void doGet(HttpServletRequest request, HttpServletResponse response
                      throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<html><body>");
        out.println("<h1>Bem-vindo ao Servlet Interativo!</h1>");
        out.println("</body></html>");
    }
}
```

Vamos praticar?

Formem cinco grupos para realizar a atividade.

Primeiro momento

15 min

- Criem um *folder* explicando o que são *applets* e *servlets* e exemplificando os dois;
- Escrevam em uma linguagem clara.

Segundo momento

10 min

Apresentem para a turma o que foi desenvolvido.



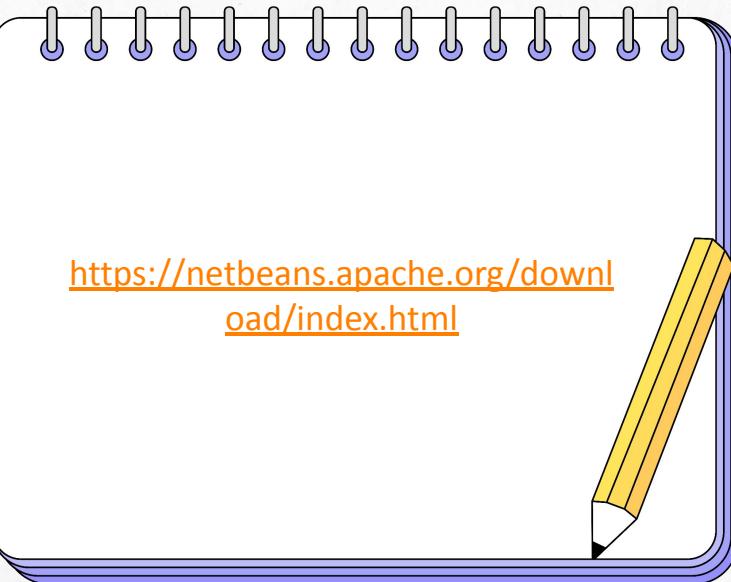
Bloco 6

Oi, faltoso!

Conte para a turma, como se estivesse falando para um amigo que faltou, um pouco sobre a sintaxe da linguagem Java, lembrando de comentar sobre *applet* e *servlet*.



Entrem nesse site!

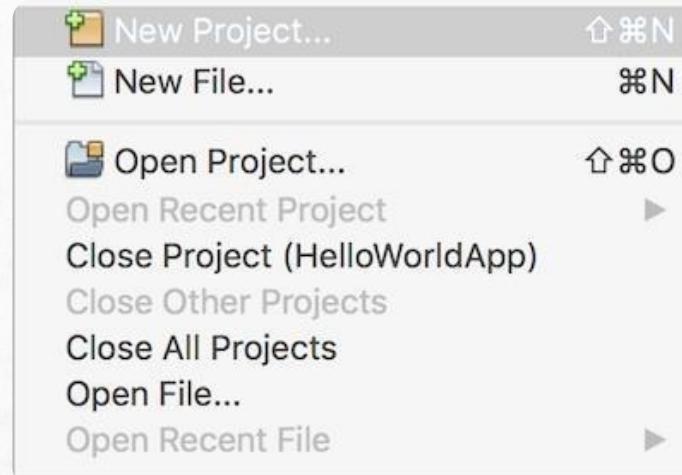


<https://netbeans.apache.org/download/index.html>

Vamos preencher juntos?

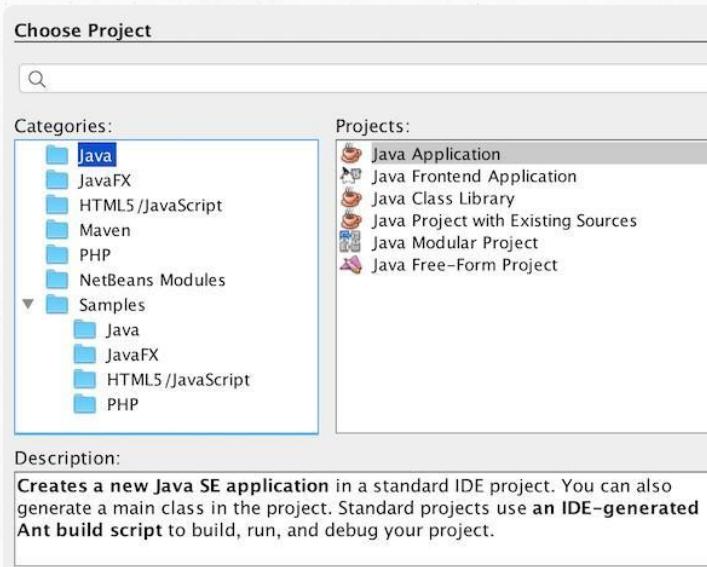
Inicie o NetBeans IDE.

Escolha o menu Arquivo > Novo Projeto



Vamos preencher juntos?

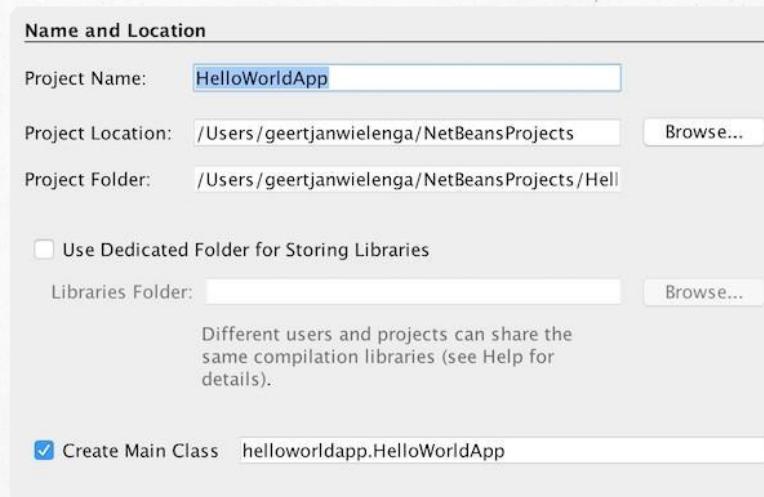
No assistente, clique em Java e selecione Aplicação Java. Em seguida, clique em Próximo.
Em Nome e Localização, adote o procedimento a seguir:



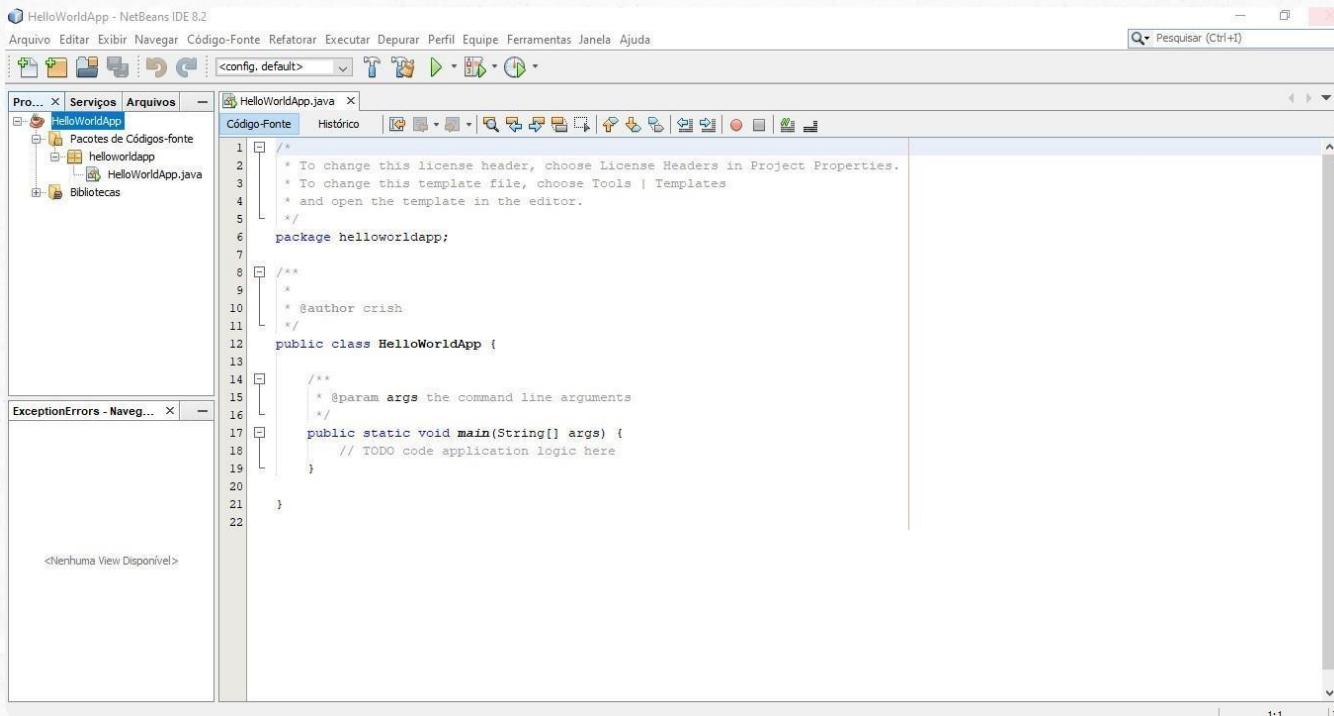
Vamos preencher juntos?

No campo Nome do Projeto, digite HelloWorldApp. Deixe desmarcada a caixa de seleção Utilizar Pasta Dedicada para Armazenar Bibliotecas.

No campo Criar Classe Principal, digite helloworldapp.HelloWorldApp. Clique em Finalizar.



A tela inicial



Copie e cole o código-fonte que o(a) professor(a) entregou e, em seguida, clique em salvar seu projeto e todas as suas classes alteradas de uma só vez.



Para executar o programa

O comando para executar o projeto ou, igualmente, F6, irá compilar e executar o código.

Escolha Executar > Executar Projeto ou o ícone 

Esse é o resultado final



The screenshot shows a terminal window within an IDE. The title bar indicates the window is titled 'Saída - HelloWorldApp (run)'. The text area displays the following output:

```
run:  
oi mundo!!!  
CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)
```

The terminal interface includes standard controls at the top: back, forward, and search buttons. On the left, there are icons for file operations like new, open, save, and close, as well as a refresh icon.

Agora, é com vocês!

Testem e explorem o aplicativo, pesquisem códigos na internet e vejam o resultado dele. Caso tenha dúvidas, tire-as com o(a) professor(a).



Fechamento

Diga, em uma palavra, o que você achou sobre a aula.



Referências Bibliográficas

PROZ EDUCAÇÃO. *Apostila de Desenvolvimento de Aplicações*. 2023