

# DA1MCTA018-13SA - Programação Orientada a Objetos - Paulo Henrique Pisani - 2021.2

[Painel](#) / [Meus cursos](#) / [POO - DA1MCTA018-13SA - 2021.2](#) / [Construtores, cópia de objetos, final](#) / [\[EP\] Arquivos multimídia](#)

Descrição

[Visualizar envios](#)

## [EP] Arquivos multimídia

**Data de entrega:** segunda, 12 Jul 2021, 23:59

**Arquivos requeridos:** ArquivoMultimidia.java, ArquivoVideo.java, ArquivoAudio.java ([Baixar](#))

**Tipo de trabalho:** Trabalho individual

## Arquivos multimídia

Neste exercício, serão utilizadas classes para representar arquivos multimídia. Haverá uma classe abstrata chamada ArquivoMultimidia e essa classe possui duas subclasses: ArquivoVideo e ArquivoAudio. A classe **ArquivoMultimidia** já está implementada na atividade e pode ser acessada na **aba Editar (o código desta classe não pode ser modificado)**. Portanto, neste exercício, devem ser implementadas apenas as subclasses ArquivoVideo e ArquivoAudio, conforme definido a seguir (todas as classes devem estar no **pacote multimidia**):

Classe **ArquivoVideo** (subclasse de ArquivoMultimidia):

- Construtor: public ArquivoVideo(String nomeArquivo, int largura, int altura) - instancia um ArquivoVideo especificando o nome do arquivo e a resolução (largura e altura).
- Método: public String toString() - sobrescreve o método toString() de Object. O retorno do método deve ser no formato "Video: %s (%d x %d)" (observe que não há acento), em que %s é o nome do arquivo e os dois inteiros indicados com %d são respectivamente largura e altura do vídeo.

Classe **ArquivoAudio** (subclasse de ArquivoMultimidia):

- Construtor: public ArquivoAudio(String nomeArquivo, boolean audioHD) - instancia um ArquivoAudio especificando o nome do arquivo e se este arquivo tem áudio HD ou não.
- Método: public String toString() - sobrescreve o método toString() de Object. O retorno do método deve ser no formato "Audio: %s (HD = %s)" (observe que não há acento), em que o primeiro %s é o nome do arquivo e o segundo %s pode assumir o valor "true" ou "false" (true se o arquivo tem áudio HD e false caso contrário).

**Importante:** Submeta apenas as classes públicas descritas no enunciado (todas devem estar no **pacote multimidia**). O programa principal já existe no sistema de correção automática. As classes submetidas não podem realizar impressão de dados ou utilizar import.

## Casos de teste

Formato dos casos de teste (que aparecem ao avaliar as classes no sistema de correção automática):

### Entrada

- quantidade de arquivos a serem instanciados
- lista de arquivos a serem instanciados no formato [tipo] [nome do arquivo] [valores de atributos]

### Saída

- resultado da chamada ao método toString() de cada arquivo instanciado

## Arquivos requeridos

ArquivoMultimidia.java

```
1  /*
2  * IMPORTANTE: NAO ALTERE ESTE ARQUIVO
3  */
4  package multimidia;
5
6  public abstract class ArquivoMultimidia {
7      private String nomeArquivo;
8
9      public ArquivoMultimidia(String nomeArquivo) {
10         this.nomeArquivo = nomeArquivo;
11     }
12
13     public final String getNomeArquivo() {
14         return this.nomeArquivo;
15     }
16 }
```

ArquivoVideo.java

ArquivoAudio.java

[VPL](#)

[◀ \[Vídeo\] Modificador final](#)

Seguir para...

[\[EP\] Perfil de usuário ▶](#)



Este é o Ambiente Virtual de Aprendizagem da UFABC para apoio ao ensino presencial e semipresencial. Esta plataforma permite que os usuários (educadores/alunos) possam criar cursos, gerenciá-los e participar de maneira colaborativa.

## Informação

[Conheça a UFABC](#)

[Conheça o NTI](#)

[Conheça o Netel](#)

## Contato

Av. dos Estados, 5001. Bairro Bangu - Santo André /SP – Brasil. CEP 09210-580.

Siga-nos



[Universidade Federal do ABC](#) - [Moodle](#) (2020)

[Obter o aplicativo para dispositivos móveis](#)