

## Tarefa 2 – Classes Compostas (21.1)

- Essa tarefa é composta por 4 classes e um bloco principal. Você deve utilizar apropriadamente as classes.
- Construa cada classe em um arquivo separado. py e o bloco principal em outro arquivo. O arquivo com o bloco principal deve ser identificado por: **Tar2\_TURMA\_PrimNome\_UltNome\_MATRICULA.py**
- Em todos os arquivos deve constar o seguinte cabeçalho:  

```
#####
#Nome completo:
#Matrícula PUC-Rio:
#Turma:
#Professor:
#Declaração de autoria: Declaro que este documento foi produzido por mim em sua totalidade,
#####
```
- Você deve dar upload em todos os arquivos
- As classes Horário e Data disponíveis:.

### Horario

Atributos: h, m e s e tempo em s do horário em relação a 0:0:0

Métodos:

<i>construtor</i>	este método constrói um objeto Horário. Há duas formas de construí-lo: a) informando h,m s com os seguintes valores default: h='', m=0, s=0. Quando h='', utiliza o horário atual. ou b) com parâmetro: tempo=qtSegundos. Quando h='', utiliza o horário atual.
<i>apresentação</i>	retorna uma string com os valores dos atributos no formato "hh:mm:ss"
-	recebe um outro Horário e retorna um novo Horário equivalente a diferença de tempo entre os horários recebidos
+	recebe um outro Horário e retorna um novo Horário equivalente a soma dos tempos dos horários recebidos
== != > <	Recebem como parâmetro um outro objeto da classe Horário, realizando a operação de comparação equivalente ao operador relacional
getSeg	retorna os segundos do horário
getMin	retorna os minutos do horário
getHora	retorna as horas do horário
clone	o objeto clona a si próprio, para isto, ele cria um novo objeto da classe Horário com os mesmos valores de atributos e retorna sua referência
setSeg	recebe um valor e altera o atributo minuto. Recalcula o tempo decorrido
setMin	recebe um valor e altera o atributo minuto. Recalcula o tempo decorrido

setHora	recebe um valor e altera o atributo minuto. Recalcula o tempo decorrido
detInterv	recebe uma quantidade de dias e calcula a data após este intervalo

Data	
Atributos: dia, mês e ano	
Métodos:	
<b>construtor</b>	este método verifica se a data está correta, caso não esteja, é configurada para valor padrão 01/01/0001
<b>apresentação</b>	retorna uma string com os valores dos atributos
-	recebe uma outra data e retorna a quantidade de dias entre as datas
+	Recebe um número de dias e retorna a data após/antes este intervalo de dias
+=	Recebe um nº de dias e altera o objeto para a data após/antes este intervalo de dias
== != > <	Recebem como parâmetro um outro objeto da classe Data, realizando a operação de comparação equivalente ao operador relacional
<b>getDia</b>	retorna o dia da data
<b>getMes</b>	retorna o mês da data
<b>getMesExtenso</b>	retorna o mês da data corrente por extenso
<b>getAno</b>	retorna o ano da data
diaDaSemana	Retorna o dia da semana da data
<b>copia</b>	o objeto clona a si próprio, para isto, ele cria um novo objeto da classe Data com os mesmos valores de atributos e retorna sua referência pelo método
<b>setDia</b>	recebe um dia e o altera se for válido
<b>setMes</b>	recebe um mês e o altera se for válido
<b>setAno</b>	recebe um ano e o altera se for válido
<b>isMesValido</b>	Retorna True se é um valor de mês válido ou False caso contrário'

isDiaValido	Retorna True se é um dia válido para o mês ou False caso contrário
isBissexto	retorna verdadeiro se o ano da data corrente for bissexto e falso caso contrário (múltiplo de 4 e (não múltiplo de 100 ou múltiplo de 400))

## Enunciado

Escreva o sistema para avaliação de filmes. Nesse exercício você deverá utilizar a classe Data a classe Horário e escrever quatro novas classes: Pessoa, Filme, CadastroDeFilmes e AssistirFilme

### A) Classe Pessoa

Toda pessoa tem: nome, e-mail, senha, data do aniversário (objeto Data)

Para criar uma pessoa no sistema devem ser fornecidos obrigatoriamente seu nome, e-mail e senha. As pessoas incluídas sem especificação da data de aniversário, devem ter como data de aniversário o valor 01/01/0000

Ao ser dado um print ou visualização da representação interna em uma pessoa, devem ser exibidos seu nome, e-mail e idade.

Uma pessoa pode:

- mostrar na tela todos os seus atributos (exibePessoa)
- alterar email, senha, data de aniversário (métodos set)
- acessar todos os seus atributos (métodos get)
- calcular a idade

### B) Classe Filme

Todo filme tem: título, duração (Horário), quantidade de visualizações, quantidade de curtidas, quantidade de não curtidas e avaliação que é um dicionário com duas chaves: somaNotas e qtNotas

Para criar um filme no sistema devem ser fornecidos obrigatoriamente seu título e duração. Os atributos quantidade de visualizações, quantidade de curtidas, quantidade de não curtidas iniciam com 0 e o atributo avaliação inicia com o dicionário {'somaNotas':0, 'qtNotas':0}

Ao ser dado um print ou visualização da representação interna em um filme, devem ser exibidos: título, duração (em minutos), quantidade de visualizações, quantidade de curtidas, quantidade de não curtidas e a nota média de avaliação (somaNotas / qtNotas)

Um filme pode:

- mostrar na tela todos os seus atributos (exibeFilme)
- alterar título e duração (métodos set)
- acessar todos os seus atributos (métodos get)
- curtir: aumenta quantidade de curtidas em 1
- naoCurtir: aumenta quantidade de não curtidas em 1
- visualizado: aumenta quantidade de visualizações em 1
- avaliacao: recebe uma nota e atualiza o valor da chave somaNota somando-lhe esta nota e o valor da chave qtNotas em 1

### C) Classe CadastroDeFilmes

O cadastro de filmes tem um dicionário de filmes, inicialmente vazio. A chave deste dicionário é o título do filme e o valor é um objeto Filme.

É possível no cadastro de filmes:

- Incluir um novo filme
- Exibir todos os filmes
- Exibir todos os filmes com avaliação acima de uma nota recebida
- Exibir um filme, dado seu título. Caso o filme não esteja no cadastro de filmes, exibir a frase: 'Filme inexistente'
- `__eq__`: recebe outro cadastro de filmes e retorna True se o dicionário de ambos tiverem o mesmo dicionário

Obs: `dic1 == dic2` retorna True se os dois dicionários tiverem os mesmos pares `chv:valor`.

---

## D) Classe AssistirFilme

Todo objeto da classe AssistirFilme tem um espectador, que é uma Pessoa e um filme, que é um objeto filme.

Ao ser dado um print ou visualização da representação interna em um objeto da classe AssistirFilme, devem ser exibidos todos os seus atributos.

Todo objeto da classe AssistirFilme, ao ser criado, atualiza a quantidade de visualizações do filme.

Todo objeto da classe AssistirFilme pode:

- Acessar seus atributos (métodos get)
  - Avaliar filme: que recebe uma nota e atualiza a avaliação do filme assistido
  - Dar like: que atualiza a quantidade de curtidas do filme assistido
  - Dar unlike: que atualiza a quantidade de likes do filme assistido
- 

Escrever um bloco principal que:

1. construa dois objetos da classe Pessoa e exiba-os
2. construa objeto `cadFilmes`, instância da classe `CadastroDeFilmes`.
3. A partir do dicionário abaixo, com alguns dos filmes indicados ao Oscar de 2021 e em quais plataformas de streaming no Brasil estão disponíveis, construa objetos da classe `Filme` e inclua-os em `cadFilmes`.

```
dOscar2021= {
    'Mulan': {'Duracao': 115, 'Plataforma': 'Disney Plus'},
    'Pinóquio': {'Duracao': 125, 'Plataforma': 'não disponível'},
    'Soul': {'Duracao': 107, 'Plataforma': 'Disney Plus'},
    'Meu pai': {'Duracao': 97, 'Plataforma': 'Now'},
    'Os 7 de Chicago': {'Duracao': 130, 'Plataforma': 'Netflix'},
    'O som do silêncio': {'Duracao': 130, 'Plataforma': 'Amazon Prime'},
    'Bela vingança': {'Duracao': 114, 'Plataforma': 'não disponível'},
    'Nomadland': {'Duracao': 110, 'Plataforma': 'não disponível'},
    'Minari': {'Duracao': 116, 'Plataforma': 'não disponível'},
    'Mank': {'Duracao': 131, 'Plataforma': 'Netflix'},
    'Judas e o messias negro': {'Duracao': 126, 'Plataforma': 'não disponível'}
}
```

*Atenção: A classe `Horário` fornecida na aula espera o tempo em segundos e a duração no dicionário está em minutos. Por exemplo, ao criar o objeto `Horario(tempo=140)`, o valor de exibição seria: 00:02:20*

4. exiba `cadFilmes`
5. construa 6 objetos da classe `AssistirFilme` onde:
  - a. 4 devem avaliar o filme
  - b. 3 devem dar like
  - c. 1 deve dar unlike
6. Exiba `cadFilmes`