

ECM251 - Linguagens de Programação I

13 - Arquivos

Prof. Murilo Zanini de Carvalho

Sumário

- Revisão dos conceitos apresentados até o momento
- Planejamento para 3 bimestre
- Serialização de dados
- Escrita e leitura de arquivos
- Dados estruturados
- Escrita e leitura de dados estruturados em arquivos

Revisão dos Conceitos Apresentados até o Momento

Programas Orientados a Objeto

- Os objetos vão se conversar através de mensagens. Essas mensagens podem ser implementadas através de métodos, mas não é obrigatório.
- Métodos são funções que manuseiam as mensagens entrantes.
- Alguns métodos podem utilizar funções para fazer trabalhos simples de decodificação, cálculo ou algo do tipo.

Messages
**objects send
messages to
one another**

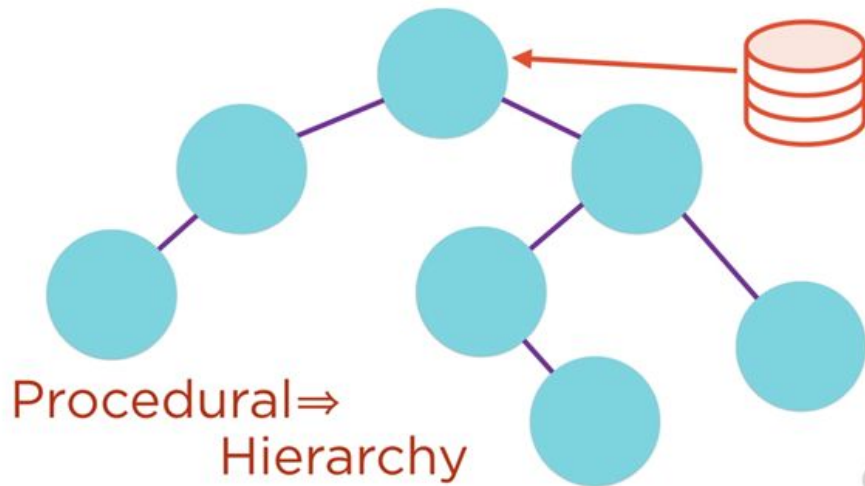
Methods
handle messages

Functions
do arbitrary work

Paradigma Orientado a Objetos

Na abordagem procedural

Data centric, e o fluxo de dados é muito importante. O sistema possui acesso a um conjunto de dados e todos os módulos alteram esses dados de alguma forma (Sanches, 2018).

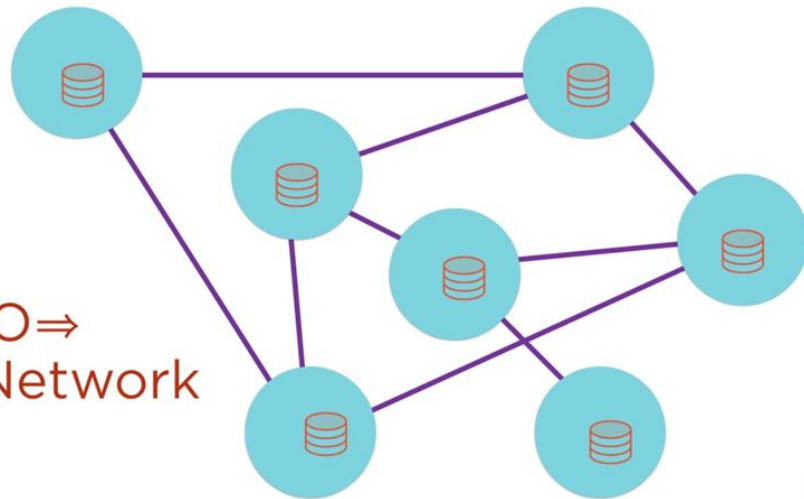


Paradigma Orientado a Objetos

Na abordagem orientada a objetos

OO não é data centric, parece um rede.
Cada objeto será responsável por gerenciar o seu próprio conjunto de dados (Sanches, 2018).

OO \Rightarrow
Network



Pilares da Orientação a Objetos

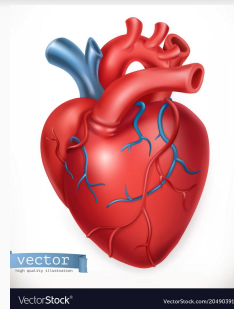


Retirado de
(<https://www.vivaxsolutions.com/images/four-pillars.png>), em
01/08/2019

Abstração

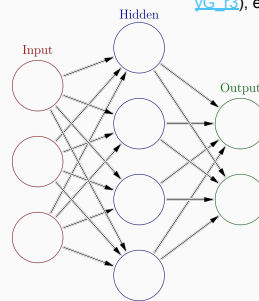
Os pontos fundamentais de um objeto ou problema são listados para representar esse objeto ou problema.

Quando essa representação fica muito complexa, ela deve ser dividida em abstrações mais simples. Cada abstração deve ter uma funcionalidade limitada, dessa forma, é possível gerenciar suas ações e seu impacto no projeto.



Retirado de
(<https://cdn2.vectorstock.com/11000x1000/03/91/human-heart-medical-internal-organs-3d-icon-vector-20490391.jpg>), em 01/08/2019

Retirado de
(https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcRSXKF4VaQV5UUXP6J-DiiQeJB-arF9dhVxsKCNa1R3Hsi_vY_r3), em 01/08/2019



Retirado de
(<https://cdn.the-scientist.com/assets/article/No/36663/llmq/15248/d305ec2a-9f5a-4894-8cd3-a7c43bb0756b-brain-640.jpg>), em 01/08/2019

Retirado de
(https://upload.wikimedia.org/wikipedia/commons/thumb/4/46/Colored_neural_network.svg/1200px-Colored_neural_network.svg.png), em 01/08/2019

Encapsulamento

Quando um garçom vai fazer um pedido para o chef, ele não deve dizer ao chefe como ele deve cozinhar, ele apenas deve informar o que foi pedido. Cada classe é responsável por suas tarefas. Ainda assim, é necessário que o garçom possa fazer o pedido para o chef.

Para essa função, existe o local onde ele deve fazer esse pedido e receber de volta do chef o prato quando ele estiver pronto. Esse conceito é o da interface pública do objeto.



Retirado de
(<https://c8.alamy.com/comp/DX8DP7/chef-at-the-pass-plating-food-michelin-starred-the-three-chimneys-DX8DP7.jpg>), em 01/08/2019

Herança

Quando as abstrações estão sendo construídas, podem existir métodos comuns em diversas classes propostas. Para retirar essa duplicação de código e para concentrar essas características comuns, é possível utilizar a herança.

Quando um objeto herda do outro, ele recebe seus métodos e atributos, tornando se também um objeto daquela classe.



Polimorfismo

Quando herdamos as características de um classe pai, alguns comportamentos podem precisar ser sobrescritos. Isso traz uma personalização para a classe filho.

Quando o método personalizado for chamado de uma instância da classe filha, ele vai ser chamado. Quando a chamada vier de uma instância da classe pai, o método original vai ser chamado.



Retirado de
(<https://i.pinimg.com/originals/d4/6b/bc/d46bbc09f471eb370ccdf30aae8ae9a9.png>), em 01/08/2019

Objetos

- Objetos são definidos pelo o que eles fazem, não pelo o que eles contêm. Eles devem ser vistos como uma caixa preta, que você pede para eles fazem algumas operações, e você não sabe qual a implementação e qual o processamento que ele irá realizar. Então podemos dizer que os objetos possuem responsabilidades, e devem realizar operações com coesão.
- Então não sabemos como o objeto funciona, mas sabemos como pedir coisas para eles, bem como o que esperar como resposta. O que existe dentro do objeto deve ser desconhecido para nós, como uma caixa preta.



Objetos - Princípio de Responsabilidade Única

Cada objeto deve possuir apenas uma responsabilidade. As duas imagens abaixo violam esse princípio.



Single Responsibility



Objetos - Acoplamento

Se você está pedindo informações para uma classe, isso pode ser indício que o acoplamento está sendo maior do que deveria.

Se precisamos realizar algum processamento com informações que o objeto possui, devemos então pedir para o objeto realizar o processamento e não pegar a informação para tal.



Ask for help,
not for information.

Don't *get()* the data.
Ask the object that
has the data to do
the work for you.

Delegation

Objetos - Isolamento

Devo ser capaz de modificar qualquer variável, tipo ou implementação da minha classe, sem que as que utilizam ela se quer notem a diferença.

You should be able to radically change the implementation of a class without impacting the clients.

Linguagem Java

Criada por James Goslin, Mike Sheridan e Patrick Naughton, em 1991 como uma linguagem de propósito geral. Em 2020, continua sendo umas das linguagens sendo mais utilizada no mercado. Para saber mais:

- [https://en.wikipedia.org/wiki/Java_\(programming_language\)](https://en.wikipedia.org/wiki/Java_(programming_language))
- <https://www.oracle.com/java/moved-by-java/>



Retirado de
(https://upload.wikimedia.org/wikipedia/en/thumb/3/30/Java_programming_language_logo.svg/141px-Java_programming_language_logo.svg.png), em 03/08/2020



Retirado de
(https://upload.wikimedia.org/wikipedia/commons/thumb/1/14/James_Gosling_2008.jpg/220px-James_Gosling_2008.jpg), em 03/08/2020

Tipos Primitivos

Data Type	Size	Description
byte	1 byte	Stores whole numbers from -128 to 127
short	2 bytes	Stores whole numbers from -32,768 to 32,767
int	4 bytes	Stores whole numbers from -2,147,483,648 to 2,147,483,647
long	8 bytes	Stores whole numbers from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
float	4 bytes	Stores fractional numbers. Sufficient for storing 6 to 7 decimal digits
double	8 bytes	Stores fractional numbers. Sufficient for storing 15 decimal digits
boolean	1 bit	Stores true or false values
char	2 bytes	Stores a single character/letter or ASCII values

Adaptado de
(https://www.w3schools.com/java/java_data_types.asp), em
03/08/2020

Definição de Classes em Java

Class

- Class is a set of *attributes* and *operations* that are performed on the attributes.

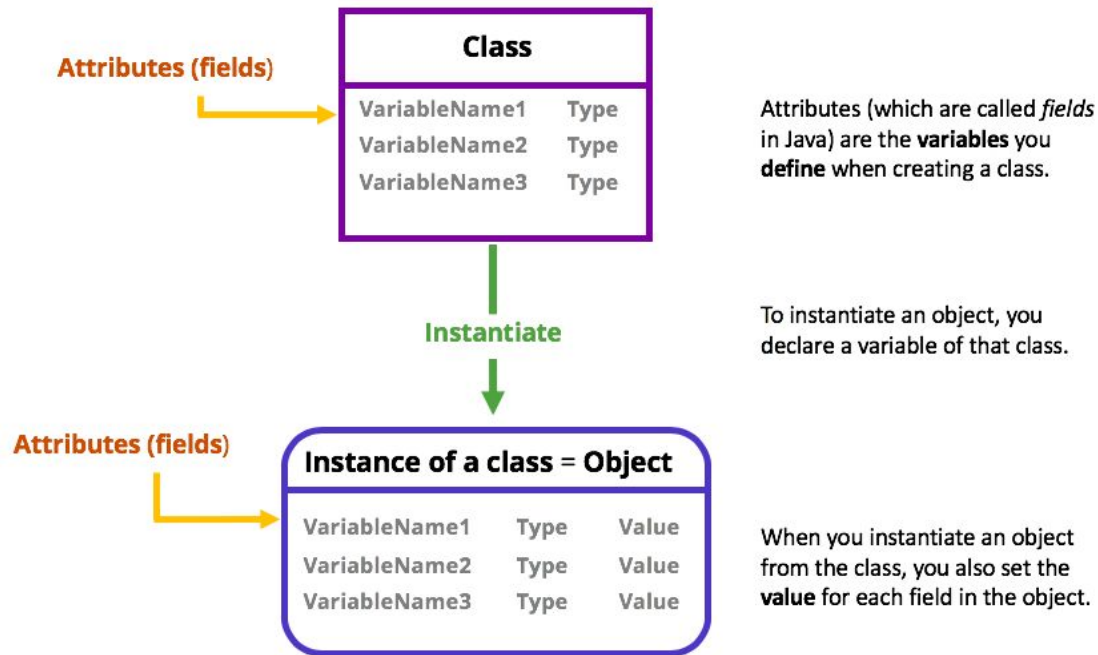
Account
accountName accountBalance
withdraw() deposit() determineBalance()

Student
name age studentId
getName() getId()

Circle
centre radius
area() circumference()

Retirado de
(<https://image.slidesharecdn.com/corejavacompletenotes-121028072655-phpapp01/95/core-java-complete-notes-contact-at-918146145674-14-638.jpg?cb=1583869278>), em 03/08/2020

Definição de Classes em Java



Retirado de
(https://user.oc-static.com/upload/2018/12/19/15452218134233_Capture%20d%E2%80%99e%CC%81cran%202018-12-19%20a%CC%80%2013.16.39.png), em 03/08/2020

Definição de Classes em Java

Classes Abstratas

```
abstract class class-name  
{  
  
non-abstract/regular methods //defined within { }  
{  
  
abstract methods(); //ending with a semicolon;  
}
```

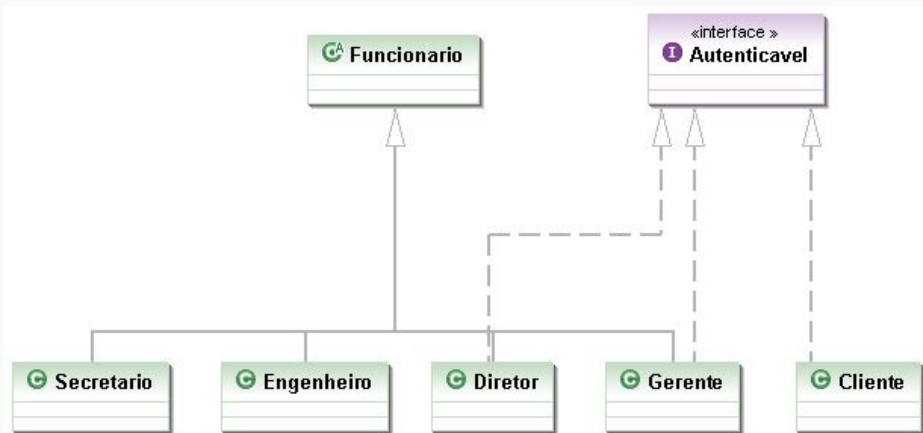
An abstract class may contain a mix of non-abstract methods and abstract methods, or only abstract methods ,or only non-abstract regular methods.

Syntax of Abstract class

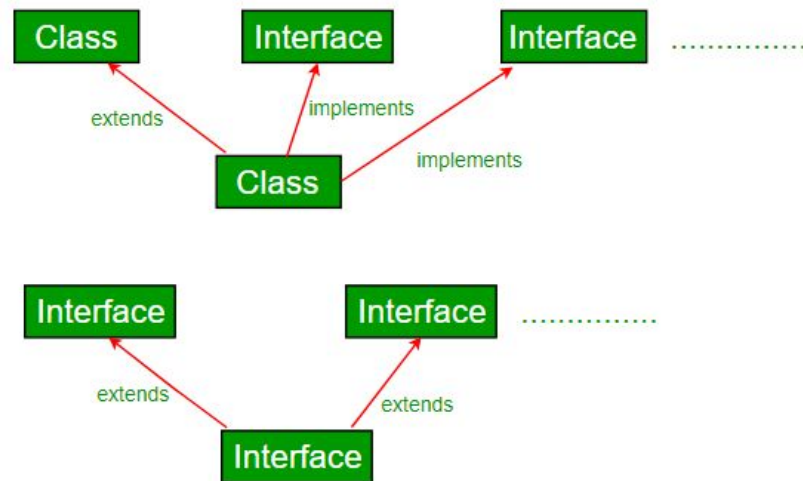
Retirado de
(<https://www.decodejava.com/java-abstract-class.png>), em
03/08/2020

Definição de Classes em Java

Interfaces



Retirado de
(https://www.caelum.com.br/apostila-java-orientacao-objetos/assets/images/interfaces/interface_autenticavel.png), em 03/08/2020



Retirado de
(<https://media.geeksforgeeks.org/wp-content/cdnuploads/extends.png>), em 03/08/2020

Definição de Classes em Java

Classes Abstratas vs Interfaces

Abstract class vs Interface (Different)

Abstract class

- To declare an abstract class, use **abstract** keyword.

```
public abstract class B{  
}
```

- A class can extend **only one** abstract class.

```
class A extends B{  
}
```

- In relationship, we say **A is B.**

Interface

- To declare an interface, use **abstract** keyword.

```
public interface B{  
}
```

- A class can implement **more than one interface.**

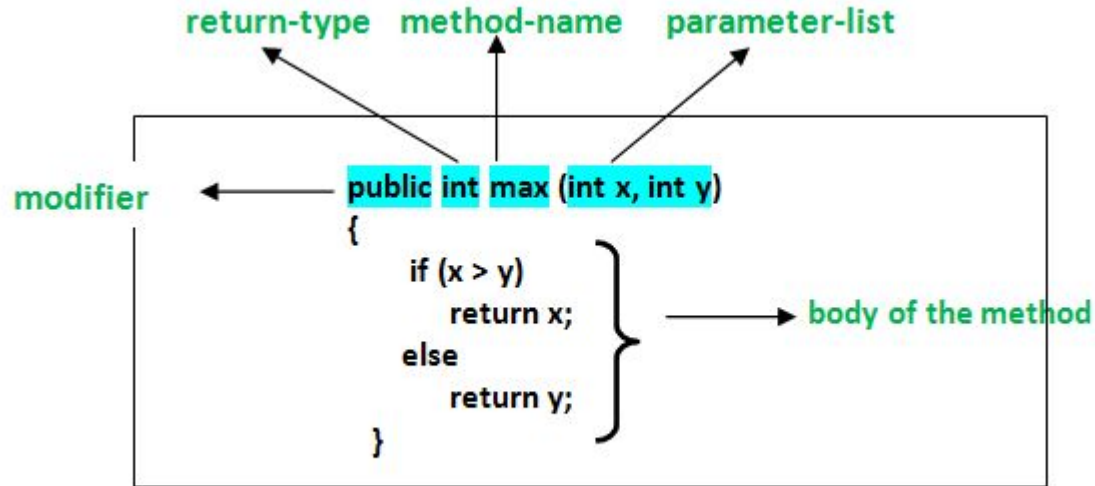
```
class A implements C, D, E{  
}
```

- In relationship, we say **A has C, D, and E.**



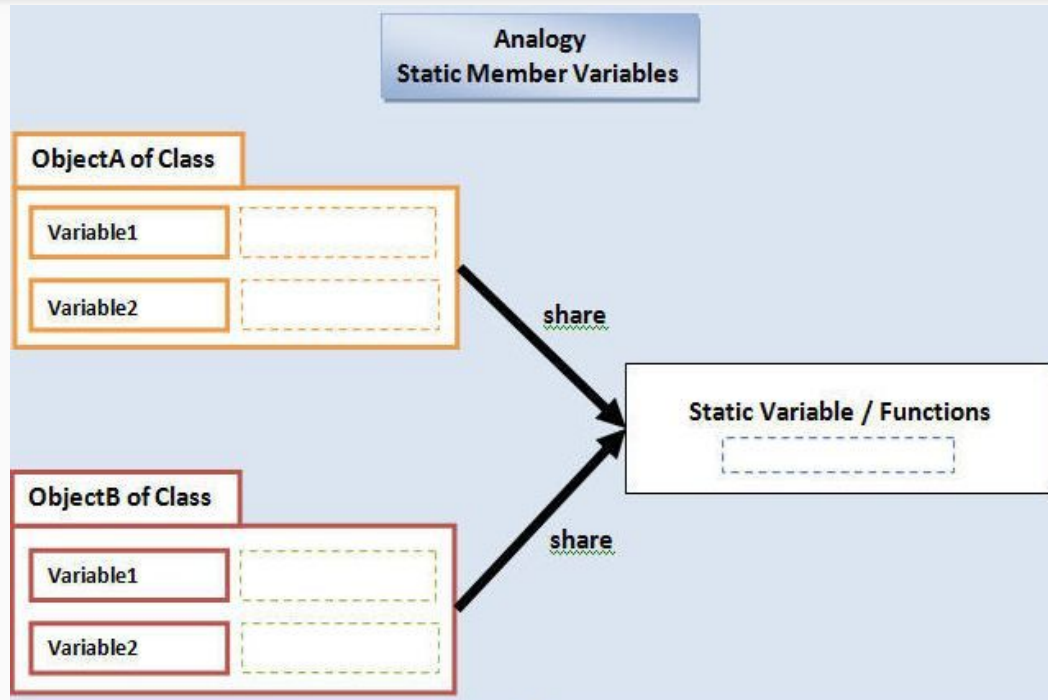
Retirado de
(<https://2.bp.blogspot.com/-5T-SEFL-cr8/V11363lQS6I/AAAAAAAAAGP0/nwA3eCtQANcmxqtrdsxqacXFKJS-uK4TQCLcB/s1600/Differenece%2Bbetween%2Binterface%2Band%2Babstract%2Bclass%2Bin%2BJava.jpg>), em 03/08/2020

Métodos



Retirado de
(<https://media.geeksforgeeks.org/wp-content/uploads/methods-in-java.png>), em 03/08/2020

Atributos e Métodos de Classe (*static*)

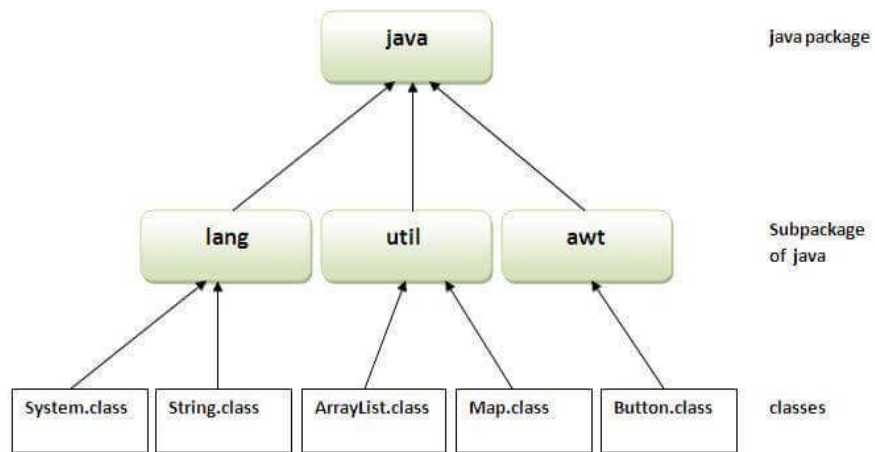


Retirado de
(<https://img.c4learn.com/2012/02/static-variable-functions1.jpg>), em 03/08/2020

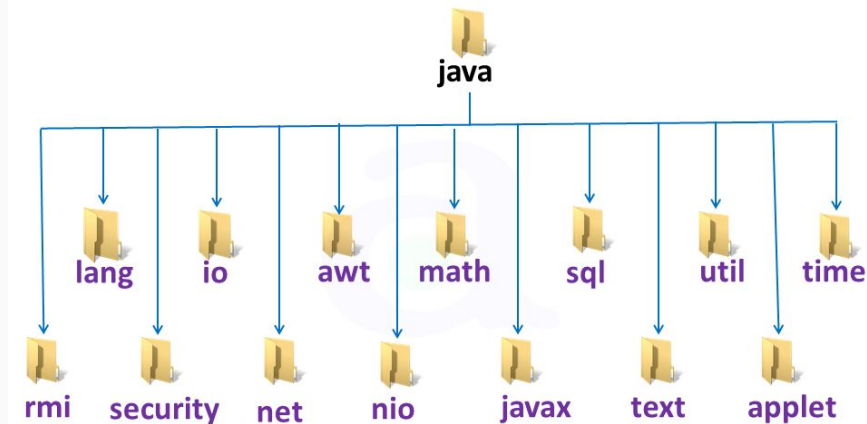
Modificadores de Visibilidade

Access Levels				
Modifier	Class	Package	Subclass	World
public	Y	Y	Y	Y
protected	Y	Y	Y	N
no modifier	Y	Y	N	N
private	Y	N	N	N

Pacotes



Retirado de
(<https://static.javatpoint.com/images/package.JPG>), em
03/08/2020



Retirado de
(<https://www.atnyla.com/library/images-tutorials/built-in-packages-in-java.PNG>), em
03/08/2020

Estrutura try/catch

```
try {  
    ...  
}  
catch (SomeException e1) {  
    ...  
}  
catch (AnotherException e2) {  
    ...  
}  
finally {  
    ...  
}
```

Code block for which we want to catch some exceptions

Each catch deals with a class of exceptions, determined by the run-time system based on the type of the argument

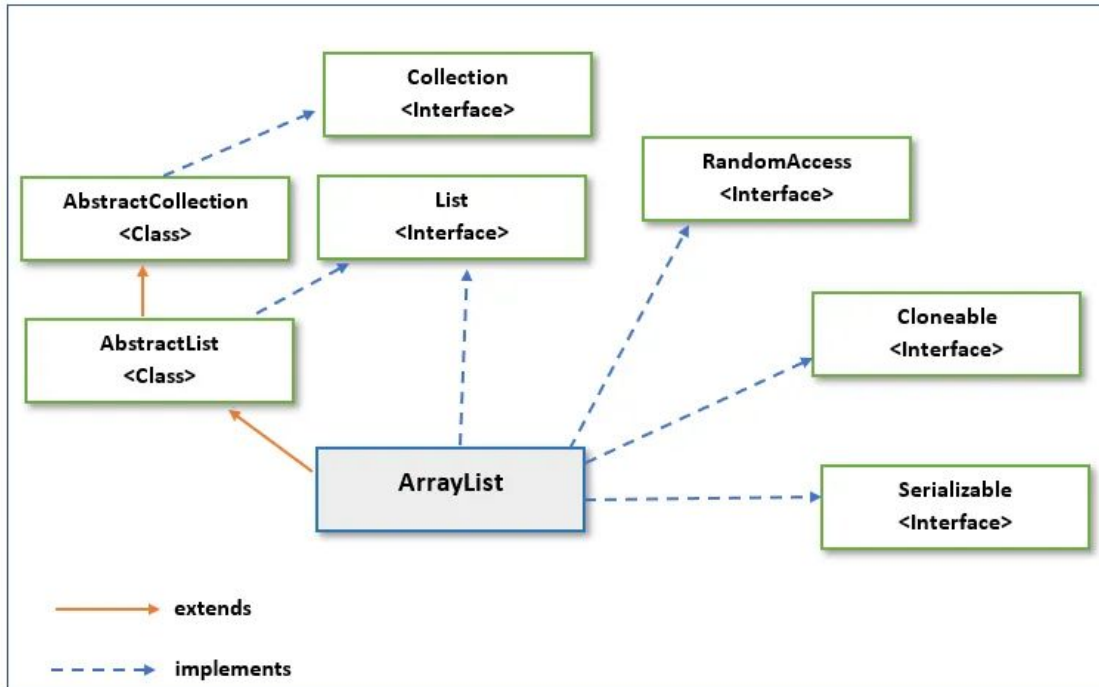
The code in finally is executed always after leaving the try-block

Array em Java

```
String[] cars = {"Volvo", "BMW", "Ford", "Mazda"};  
System.out.println(cars[0]);  
// Outputs Volvo
```

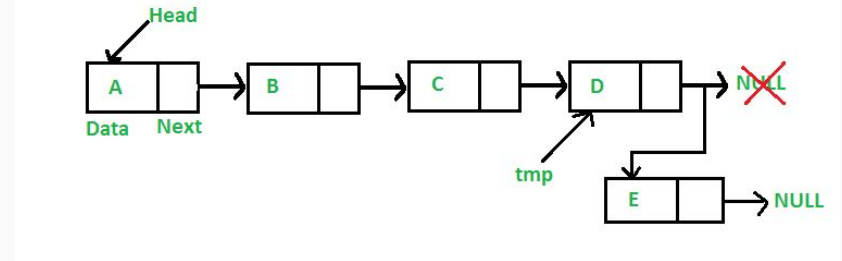
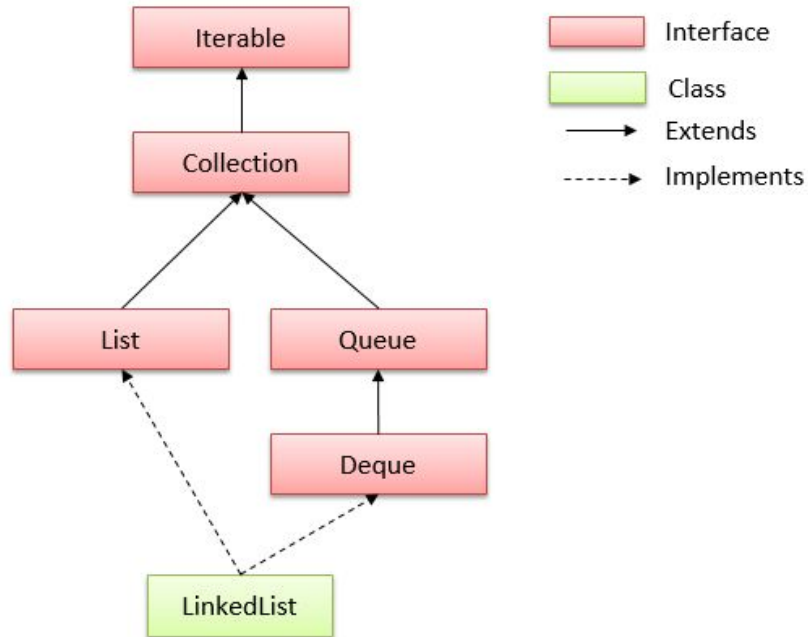
Adaptado de (https://www.w3schools.com/java/java_arrays.asp), em 03/08/2020

Coleções em Java - ArrayList



Retirado de
(<https://i0.wp.com/codippa.com/wp-content/uploads/2020/04/ArrayList-class-hierarchy.png?resize=762%2C474&ssl=1>), em 03/08/2020

Coleções em Java - LinkedList



Retirado de
(https://www.geeksforgeeks.org/wp-content/uploads/gg/2013/03/LinkedList_insert_list.png), em 03/08/2020

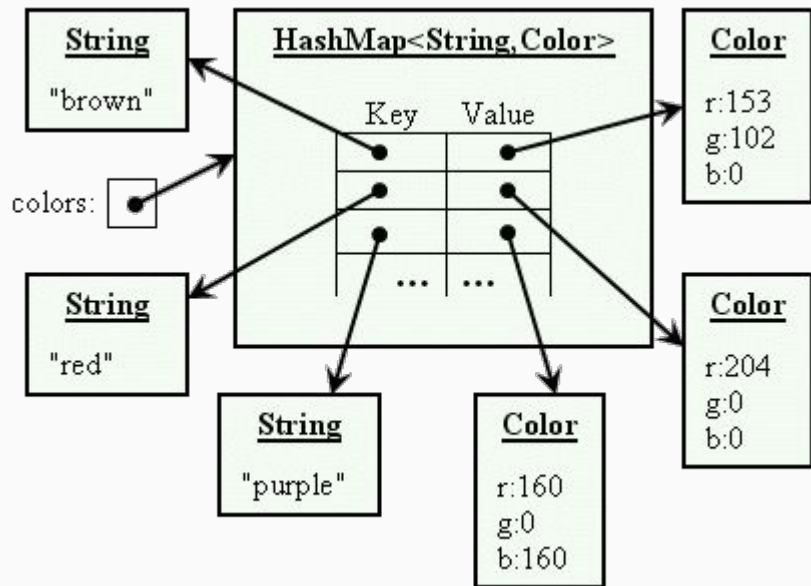
Retirado de
(<https://howtodoinjava.com/wp-content/uploads/2018/10/LinkedList-Hierarchy.png>), em 03/08/2020

Coleções em Java - HashMap

```
Map<Integer,String> map = Map.of(1, "A", 2, "B", 3, "C");
```



key	value
1	A
2	B
3	C



Coleções em Java - HashSet

	HashSet	HashMap
Implements	Set	Map
Duplicates	No	Yes, Duplicate values but no duplicate keys
Dummy values	Yes	No
Objects required during add operation	1	2
Adding or Storing mechanism	HashMap object	Hashing technique
Faster	HashSet is slower than Hashmap	HashMap is faster than HashSet
Insertion method	Add()	Put()
Null	Have a single null value	Single null key and any number of null values

Retirado de
(<https://media.geeksforgeeks.org/wp-content/uploads/hashsetVshashmap.png>), em
03/08/2020

Enumerações

Java Enum

1. Enum in Java are type safe.
2. You can define Constructor in Java Enum.
3. Enum constants are implicitly static and final.
4. Enum can implement interface in Java.
5. Enum has implicit values() method which return all enum constants in array.
6. You can use Enum in switch case like int, String data type.
7. You can compare two Enum constant using both == and equals() method.

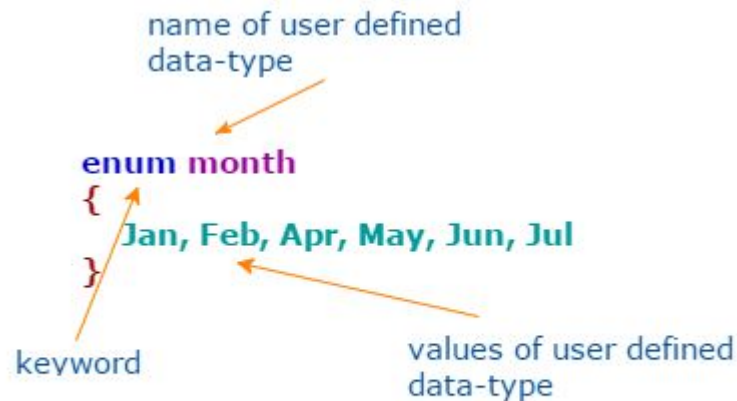
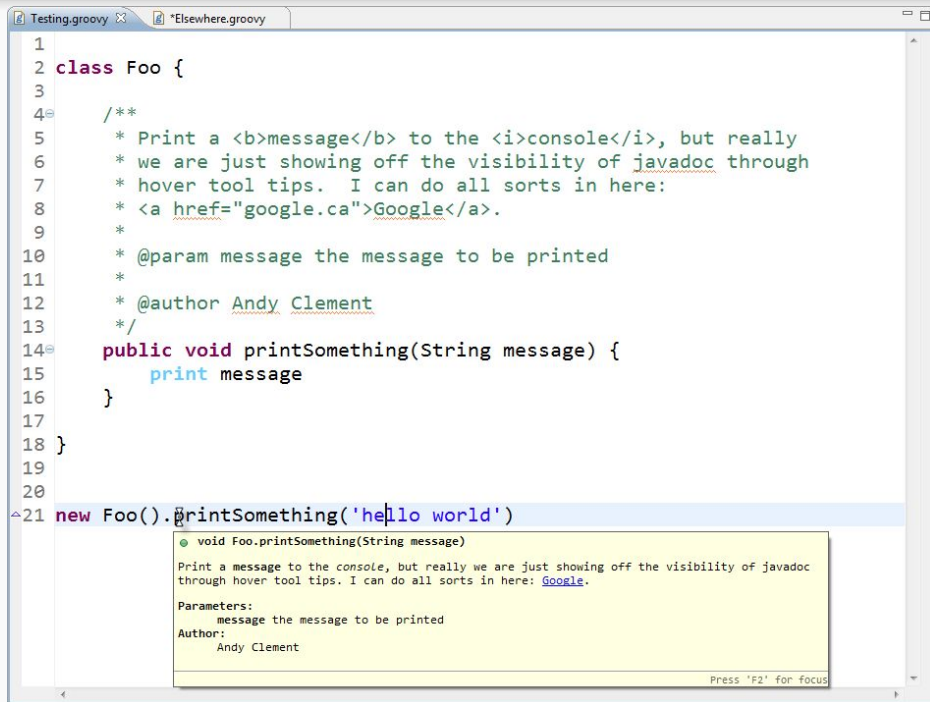


Fig: Java Enumeration

Retirado de
(<https://www.w3schools.in/wp-content/uploads/2016/09/Java-Enumeration.png>), em 03/08/2020

Javadoc



The screenshot shows an IDE window with two tabs: "Testing.groovy" and "Elsewhere.groovy". The "Testing.groovy" tab is active, displaying the following code:

```
1
2 class Foo {
3
4     /**
5      * Print a <b>message</b> to the <i>console</i>, but really
6      * we are just showing off the visibility of javadoc through
7      * hover tool tips. I can do all sorts in here:
8      * <a href="google.ca">Google</a>.
9      *
10     * @param message the message to be printed
11     *
12     * @author Andy Clement
13     */
14     public void printSomething(String message) {
15         print message
16     }
17 }
18
19
20
21 new Foo().printSomething('hello world')
```

A hover tooltip is visible over the `printSomething` method call on line 21. The tooltip contains the following information:

- Signature: `void Foo.printSomething(String message)`
- Description: Print a message to the console, but really we are just showing off the visibility of javadoc through hover tool tips. I can do all sorts in here: [Google](#).
- Parameters: message the message to be printed
- Author: Andy Clement

At the bottom of the tooltip, it says "Press 'F2' for focus".

Retirado de
(https://4.bp.blogspot.com/_gFXDbWLX580/SylW5a2b6sI/AAAAAAAAEYU/MVv7jeUFA6o/s1600/javadocs.png),
em 03/08/2020



CODE TIME!

CODE TIME

Elabore um sistema que permita o usuário cadastrar os consoles e os jogos que ele possui. Cada jogo deve estar representado com o seu status (Não Iniciado, Jogando, Não Acabado, Finalizado e Platinado!). O usuário pode mudar o status de cada um de seus jogos.

Planejamento para o 3 Bimestre

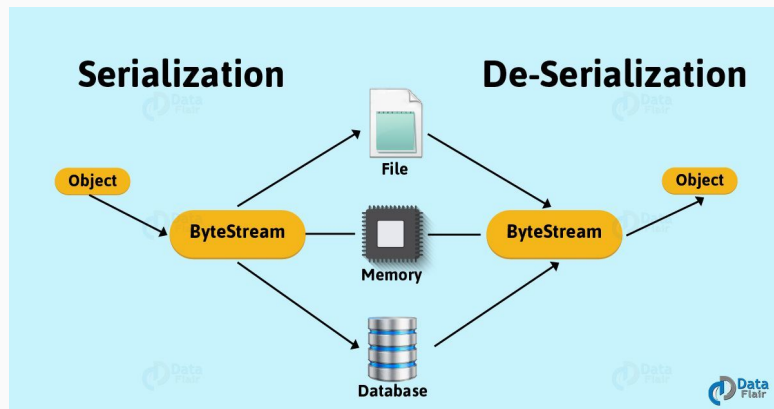
Tópicos que Serão Abordados

- Retomada. Arquivos. Conversão de JSON.
- JavaFx.
- Banco de Dados - SQLite.
- Banco de Dados - MySQL.
- Exercícios.

Serialização de Dados

Serialização de Dados

- Processo que permite transformar um objeto em sua representação em bytes.
- Possibilita armazenar, transmitir e recuperar objetos da aplicação, mantendo o seu estado atual (valores de suas variáveis de instância).



Retirado de
(<https://data-flair.training/blogs/wp-content/uploads/sites/2/2018/02/serialize-de-serialize-java-01.jpg>), em 31/07/2020

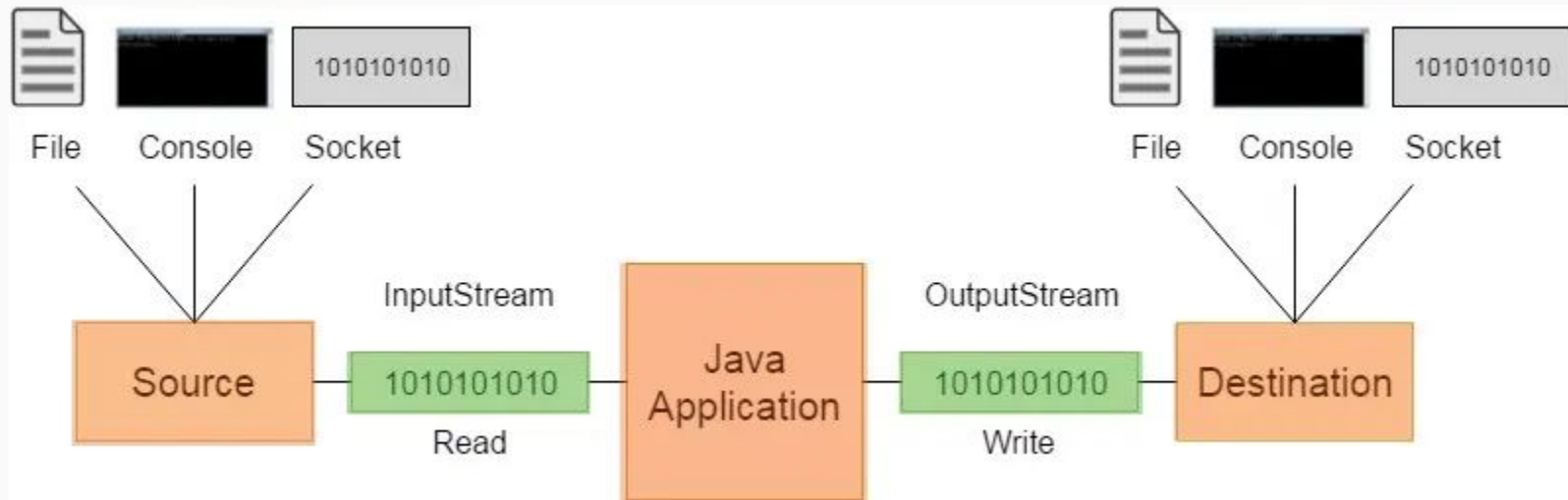
Serialização de Dados

- Os objetos que serão serializados, precisam implementar a interface Serializable.
- Esta interface não possui nenhum método que precise ser sobrescrito, ela apenas funciona como um marcador para o compilador, indicando que a classe pode ser convertida em sua representação em bytes.
- ATENÇÃO: Quando a classe possui algum tipo definido pelo usuário (objetos) dentro dela, todos eles devem implementar a interface, caso contrário, aquele objeto não terá seus atributos serializados.

Serialização de Dados - HowTo

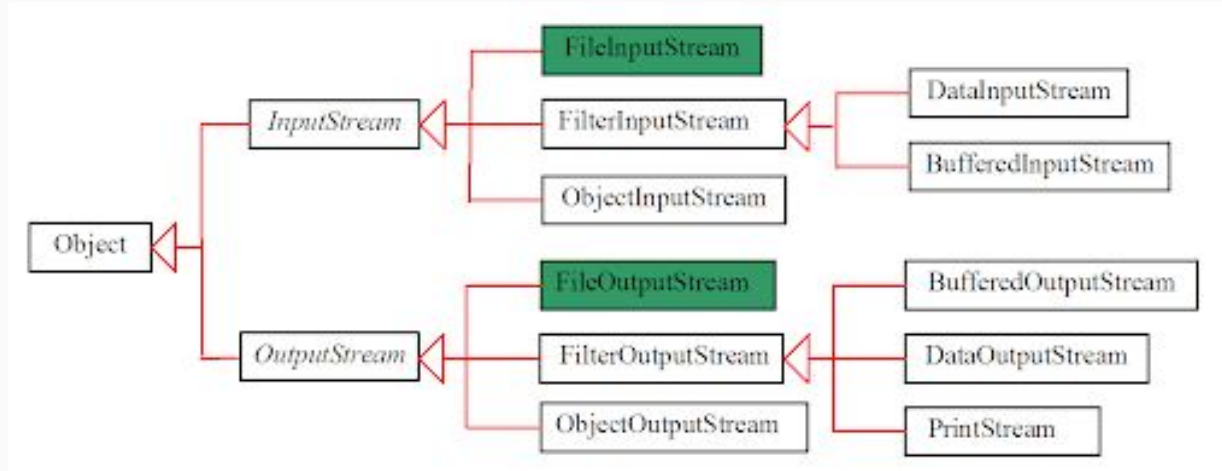
- Os objetos que serão serializados, precisam implementar a interface Serializable.
- Esta interface não possui nenhum método que precise ser sobrescrito, ela apenas funciona como um marcador para o compilador, indicando que a classe pode ser convertida em sua representação em bytes.
- ATENÇÃO: Quando a classe possui algum tipo definido pelo usuário (objetos) dentro dela, todos eles devem implementar a interface, caso contrário, aquele objeto não terá seus atributos serializados.

Serialização de Dados - HowTo



Retirado de
(<https://i0.wp.com/mytoshika.com/blog/wp-content/uploads/2019/07/java-io-flow.jpg?fit=796%2C253&ssl=1>), em 31/07/2020

Serialização de Dados - HowTo



Retirado de

(<https://lh3.googleusercontent.com/proxy/C1eXCdFSCZOJEN0TC3abKBj-6i19Z2vnUjki-zOkA5abJoflaYP2hMKtdl3w6GxbsNMxMmi3p7my8pKCzhbcjoT-u8i0cytmh-2BbqtwRV7W0vzR9-4004f4wA>), em 31/07/2020

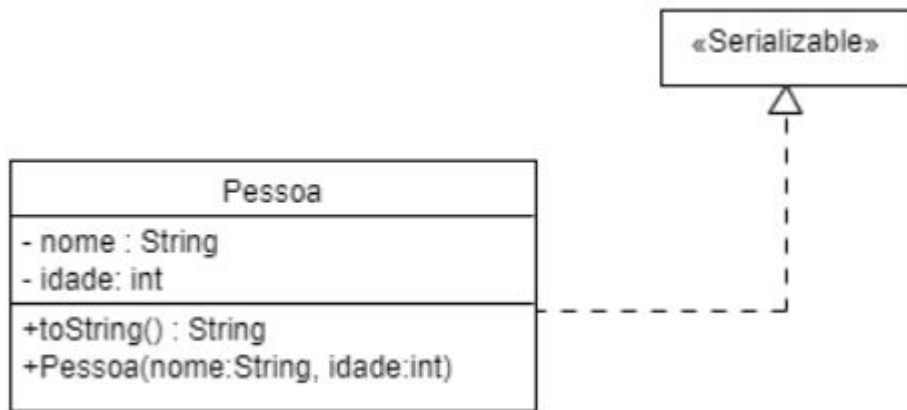
Serialização de Dados - Mais Referências

- <https://docs.oracle.com/javase/7/docs/api/java/io/Serializable.html>
- <https://www.devmedia.com.br/introducao-a-serializacao-de-objetos/3050>
- <https://www.devmedia.com.br/use-a-serializacao-em-java-com-seguranca/29012>



CODE TIME!

CODE TIME



UML:

- <https://www.umlet.com/>
- <http://www.umletino.com/>
- <https://pt.wikipedia.org/wiki/UML>
- <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/>

Escrita e Leitura de Arquivos

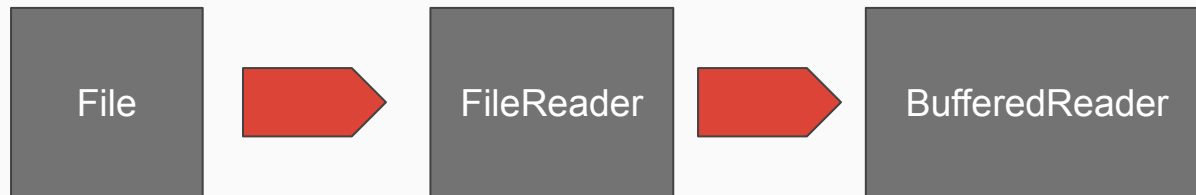
Escrita e Leitura de Arquivos

- Serializar objetos é uma forma rápida e simples de compartilhar dados do estado desse objeto. Contudo, essas informações ficam restritas a aplicações Java.
- Para escrever/ler arquivos de outros sistemas é preciso utilizar arquivos de texto/binários padrões.
- Podemos escrever nossos dados dentro dos arquivos de forma similar com o que foi realizado na etapa de serialização.

Escrita e Leitura de Arquivos

- Serializar objetos é uma forma rápida e simples de compartilhar dados do estado desse objeto. Contudo, essas informações ficam restritas a aplicações Java.
- Para escrever/ler arquivos de outros sistemas é preciso utilizar arquivos de texto/binários padrões.
- Podemos escrever nossos dados dentro dos arquivos de forma similar com o que foi realizado na etapa de serialização.

Leitura de Arquivos



<https://static.makeuseof.com/wp-content/uploads/2011/01/Format07.png>



<https://image.flaticon.com/icons/png/512/226/226777.png>

Escrita de Arquivos



<https://image.flaticon.com/icons/png/512/226/226777.png>



<https://static.makeuseof.com/wp-content/uploads/2011/01/Format07.png>

Escrita e Leitura de Arquivos

- https://www.w3schools.com/java/java_files.asp
- <https://www.geeksforgeeks.org/file-handling-java-using-filewriter-filereader/>
- <https://www.techiedelight.com/how-to-read-a-file-using-bufferedReader-in-java/>

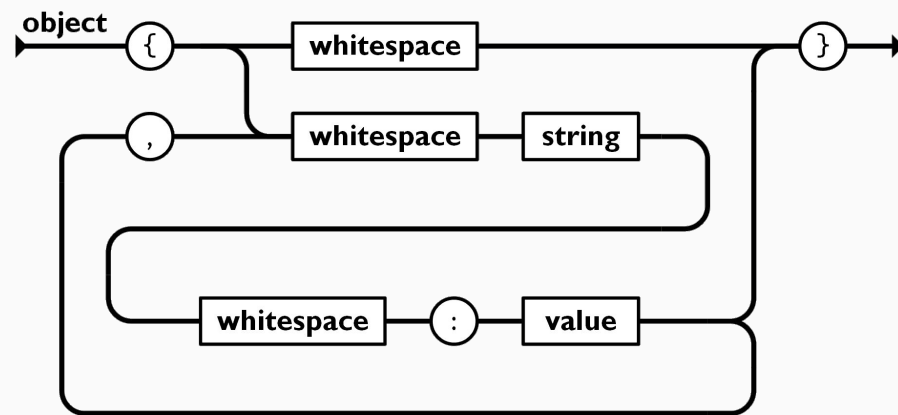


CODE TIME!

Dados Estruturados

JSON

- Estrutura de arquivo/mensagem do tipo chave:valor.
- Um dos formatos mais utilizados para troca de mensagens entre serviços e clientes.
- Mais informações:
 - <https://www.json.org/json-en.html>
 - <https://github.com/stleary/JSON-java>



<https://www.json.org/img/object.png>



CODE TIME!

Escrita e Leitura de Dados Estruturados em Arquivos

Adicionar biblioteca para manipulação de Arquivos JSON

- Acessar a URL:
 - <https://github.com/stleary/JSON-java>
- Fazer o download do arquivo jar.

README.md

JSON in Java [package org.json]

maven-central v20200518

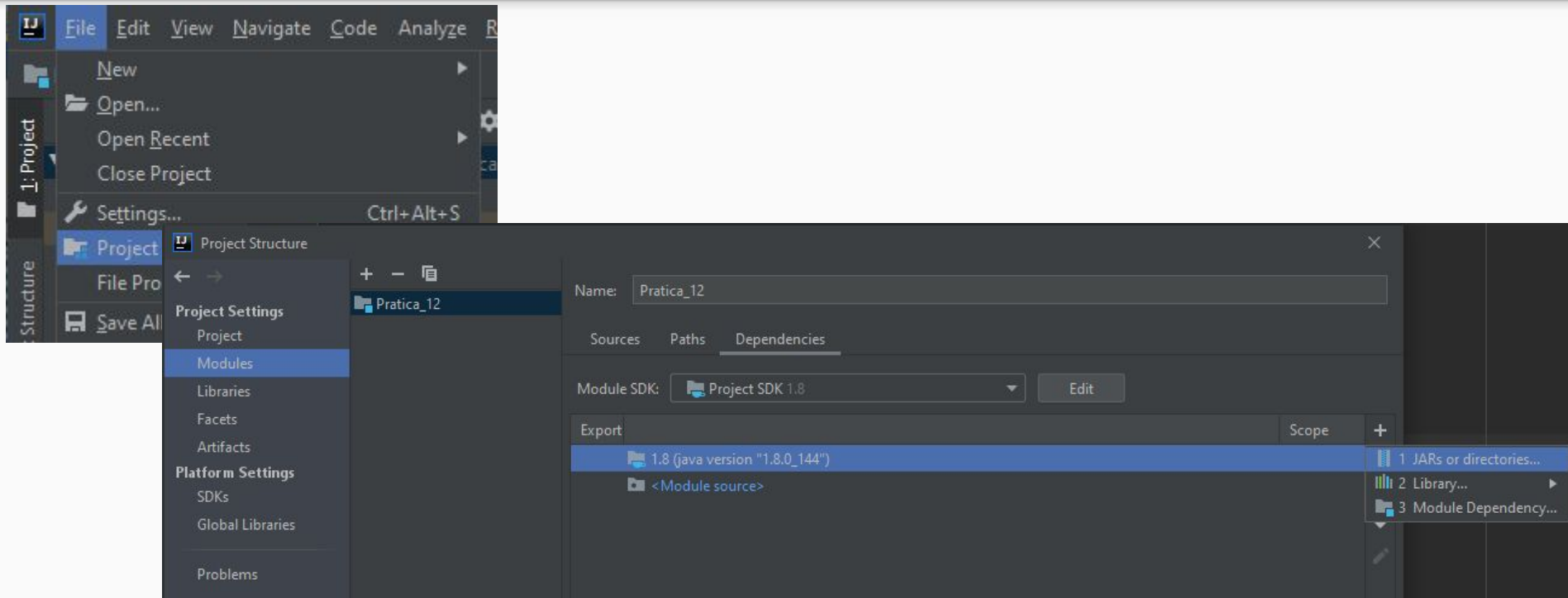
[Click here if you just want the latest release jar file.](#)

Overview

JSON is a light-weight language independent data interchange format.

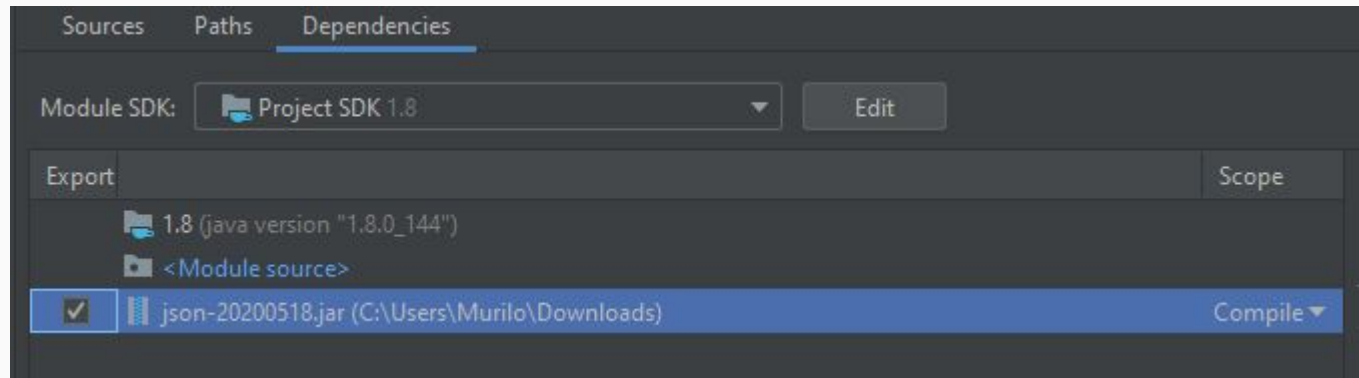
The JSON-Java package is a reference implementation that demonstrates how to parse JSON documents into Java objects and how to generate new JSON documents from the Java classes.

Adicionar biblioteca para manipulação de Arquivos JSON



Adicionar biblioteca para manipulação de Arquivos JSON

- Apontar o caminho para o download do arquivo JAR do pacote org.json.





CODE TIME!