

# NoSQL Databases

PL01 – Installations, Introduction to  
Docker, Introduction to Oracle

**Teacher:** Cristiana Neto

**Email:** [cristiana.neto@algoritmi.uminho.pt](mailto:cristiana.neto@algoritmi.uminho.pt)

**Office hours:**

Friday 10h–11h



# Summary

1

Introduction to Docker

2

Docker Installation and Configuration

3

Installation of the first container

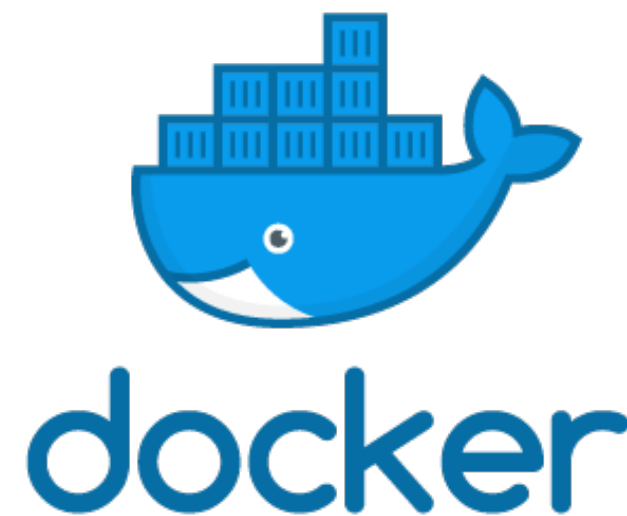
3

Introduction to Oracle

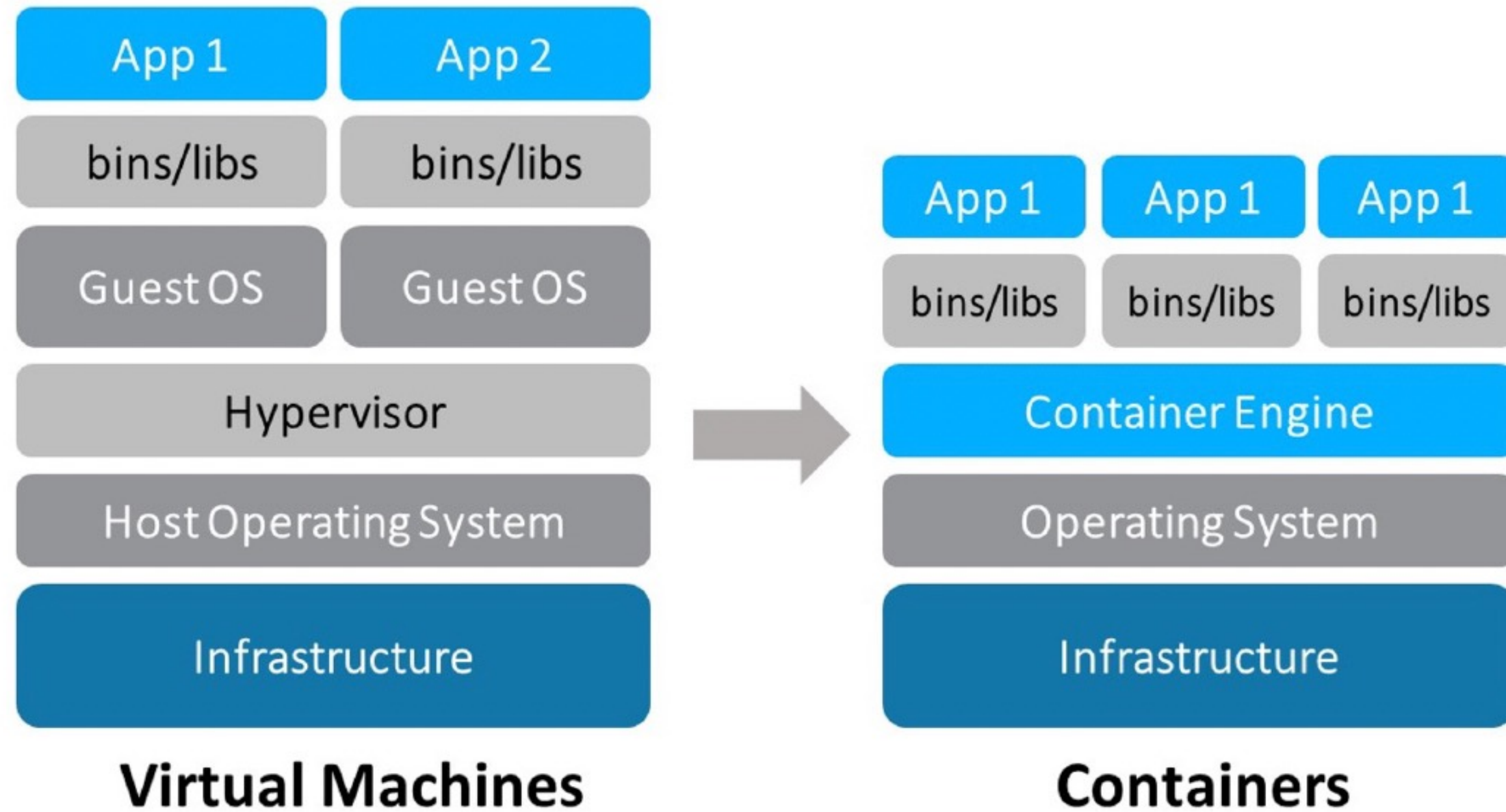
# Docker

## ➔ What is it?

Docker is an Open Source technology that allows you to build, run, test and deploy distributed applications inside software containers. It allows you to "package" a piece of software, containing everything necessary for its execution: code, runtime, tools, libraries, etc. Docker allows you to deploy applications quickly, reliably and stably, in any environment.



# Docker vs VM



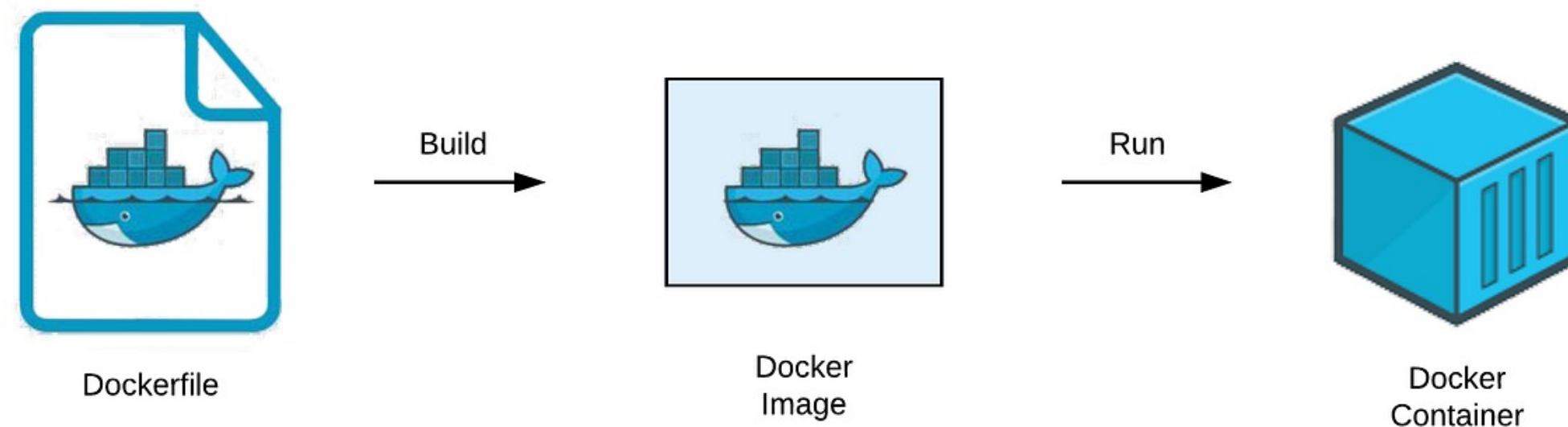
# Docker vs VM

	Docker	Virtual Machine
OS	OS shared across containers	New OS for each MV
Safety	Less secure because the operating system and kernel are shared	More secure as VMs don't share the operating system
Performance	Fast performance even with multiple containers	More virtual machines equate to less stable performance
Start-up time	Fast (seconds)	Slow (minutes)
Memory Requirements	Lightweight	Requires a lot of memory
Storage Needs	Usually megabytes	Usually gigabytes
Portability	Easy to deploy in different environments	Difficult to port a VM to another system

# Docker

## ➔ Containers

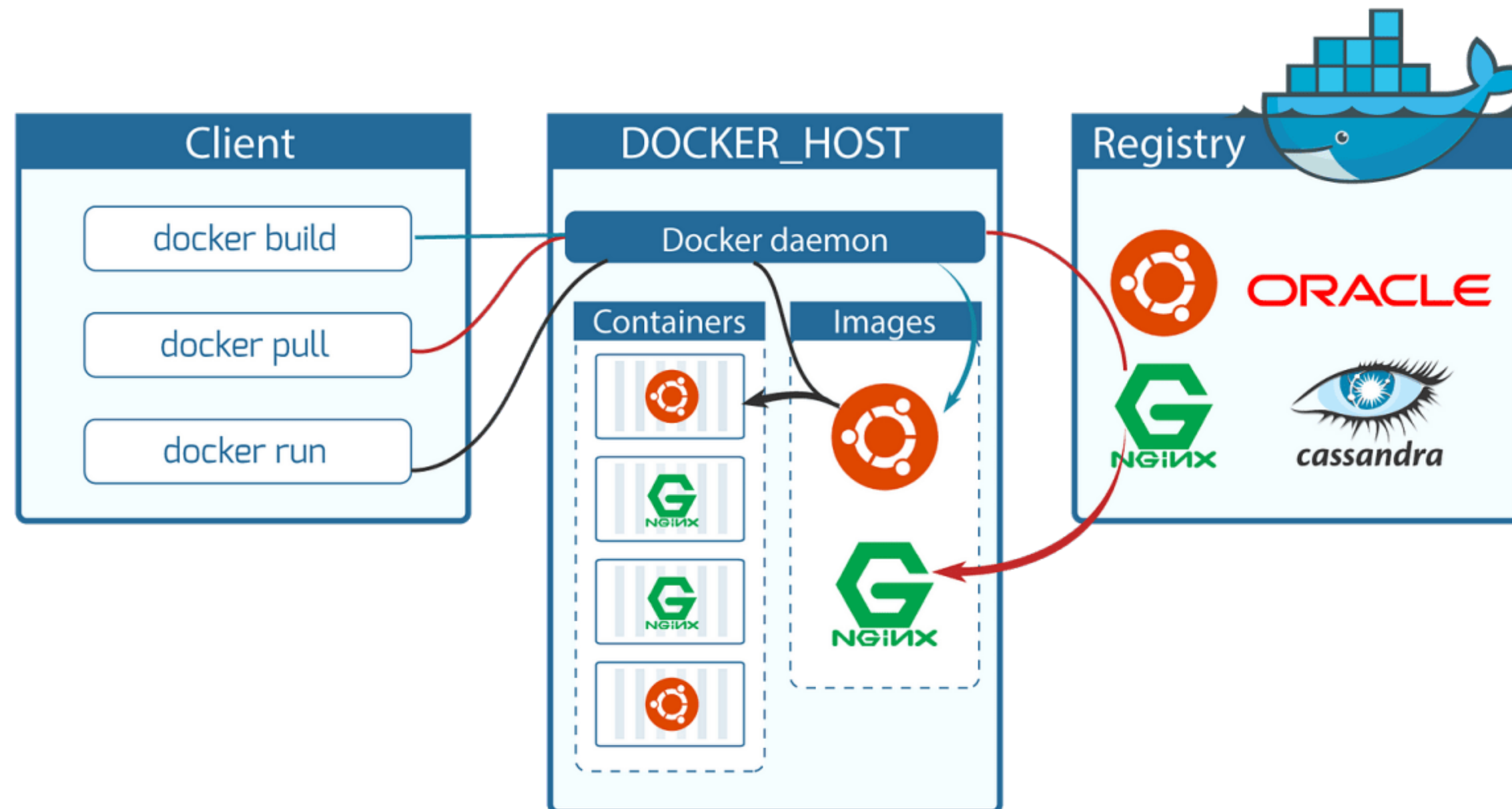
A container is a software “package” with all the dependencies needed to run a specific application. All configurations and instructions for starting or stopping containers are dictated by the Docker image. Every time a user runs an image, a new container is created.



# Docker

➔ How does it work?

## DOCKER COMPONENTS



# Docker

## ➔ Why?

- Versatility in the installation of databases to be used;
- Possibility to reverse the installation by removing the containers;
- Reduced installation time, DB configuration for different existing OS.



# Docker

## ➔ Links úteis

Getting started:

<https://docs.docker.com/get-started/>

Installation:

<https://docs.docker.com/engine/install/>

Repository:

<https://hub.docker.com>

# Exercício

## ➔ Installation of the first container

1. Install Docker;
2. Install the "helloworld" container:  

```
$ docker pull hello-world
```

```
$ docker run hello-world
```
3. Validate existing images in your Docker installation:  

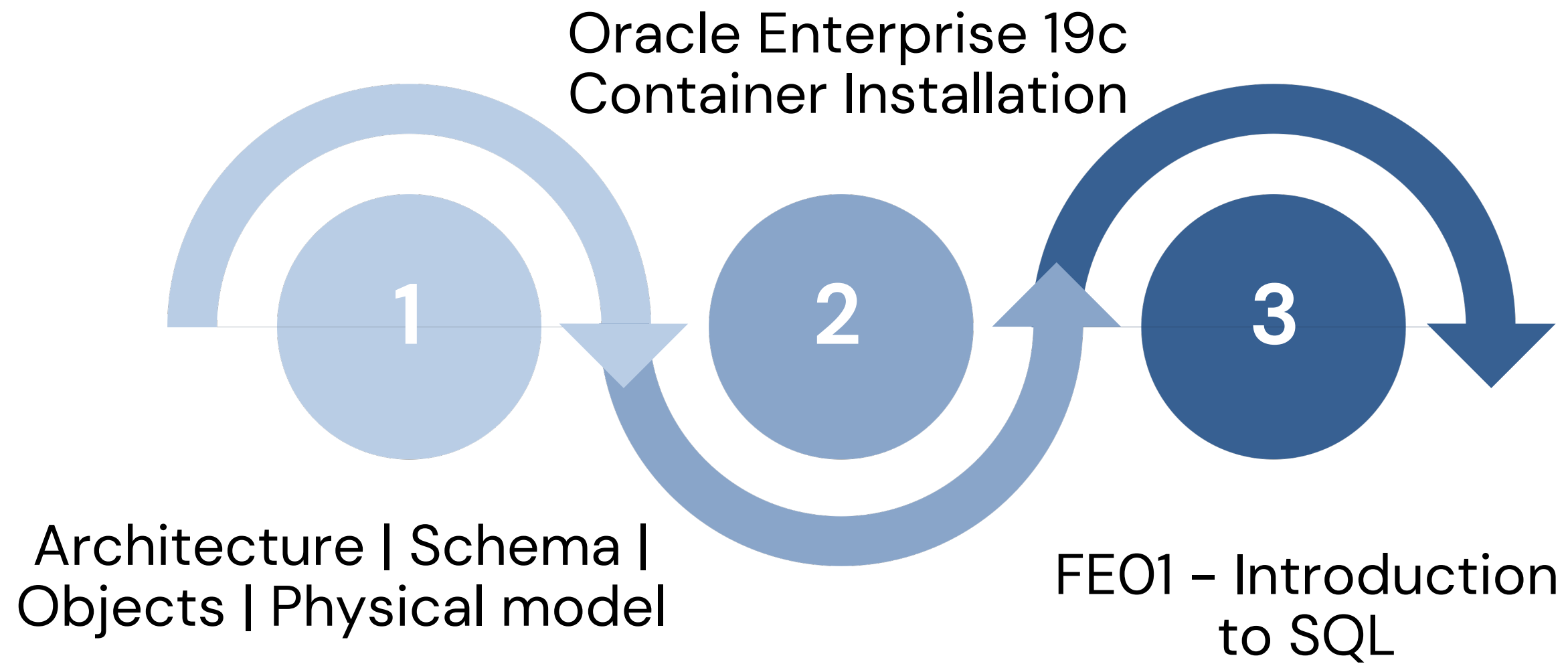
```
$ docker image ls
```

# Exercício

## ➔ Installing a graphical manager for Docker:

1. Search for the installation manual of the "Portainer" container in Docker Hub (validate the most current version);
2. Install the respective container;
3. Access the url dedicated to Portainer via the web to validate the installation.

# Oracle



# Oracle Enterprise 19c

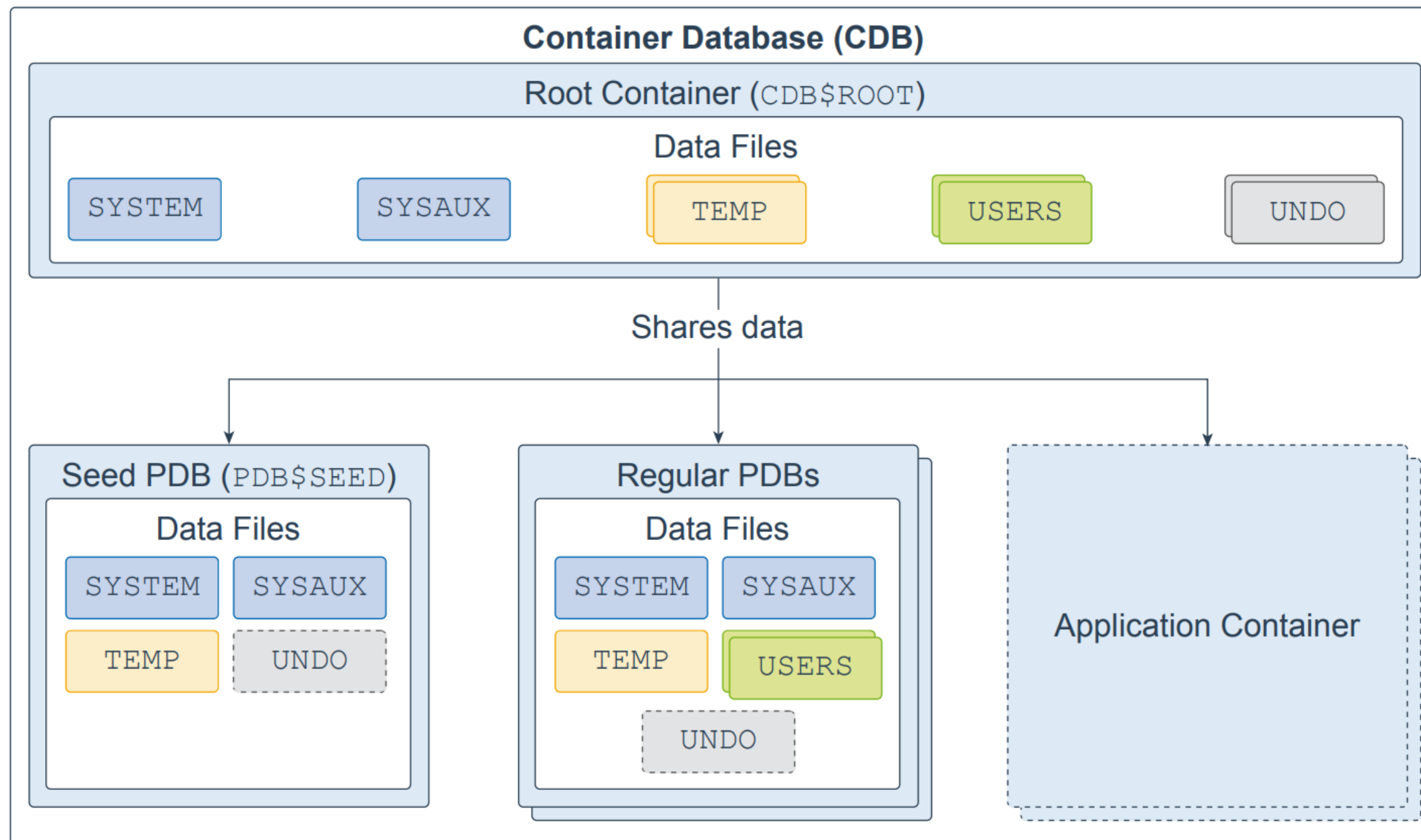
Oracle Database is a relational database management system (RDBMS) developed by Oracle Corporation. It is one of the most popular and widely used database systems in the world, known for its scalability, reliability, and robustness.



**Oracle Database 19c** is the current long-term release and provides the highest level of release stability and the longest period of time for support and bug fixes.

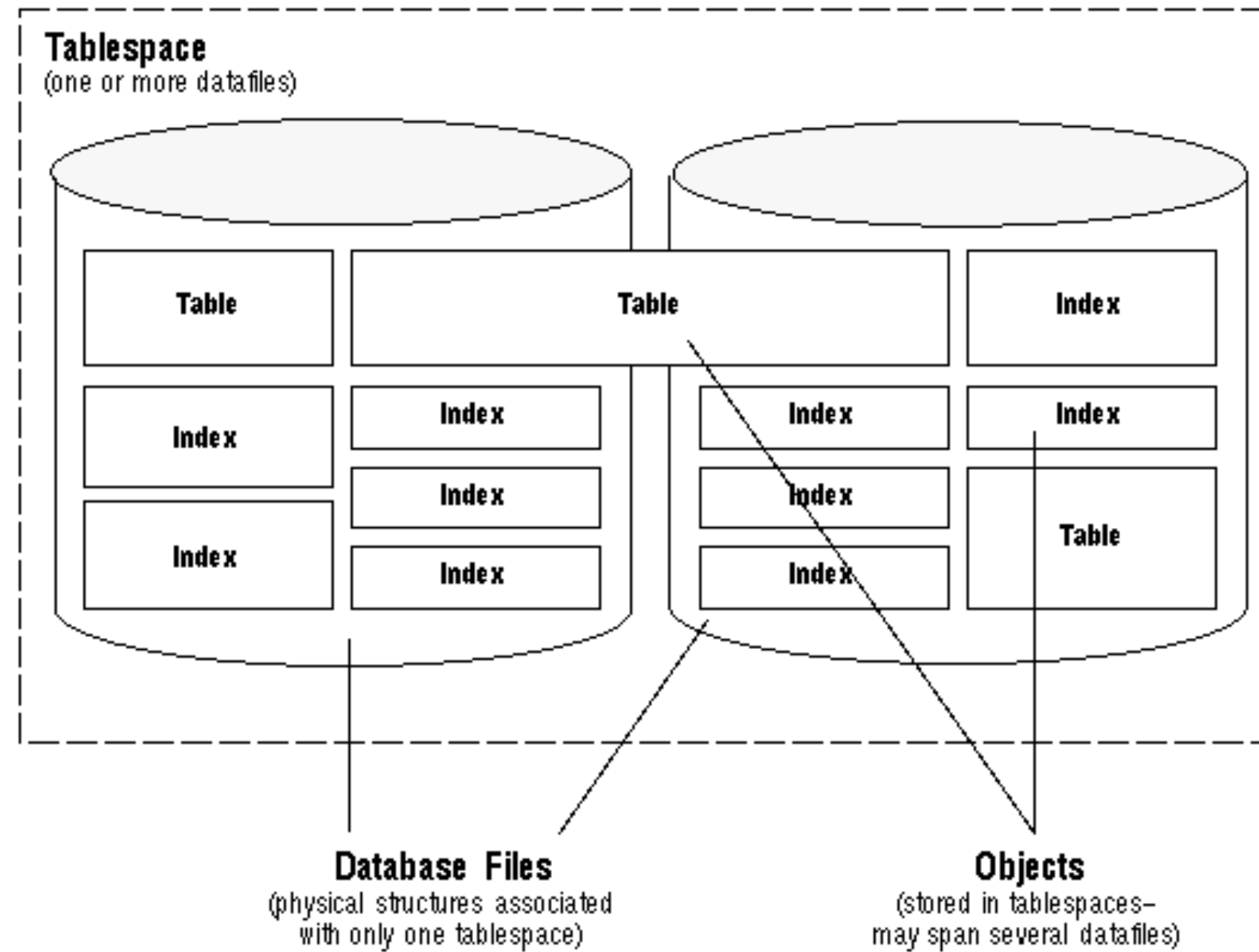
# Oracle Enterprise 19c

## ➔ Architecture



# Oracle Enterprise 19c

## ➔ Architecture



# Oracle Enterprise 19c

## ➔ Schema

### Criar Datafiles:

```
CREATE TABLESPACE uminho_tables DATAFILE 'UMINHO_FILES_01.dbf' SIZE 500m;
```

### Create User:

```
CREATE USER uminho IDENTIFIED BY "uminho2020" DEFAULT TABLESPACE uminho_tables QUOTA  
UNLIMITED ON uminho_tables;
```

### Permissions and Roles:

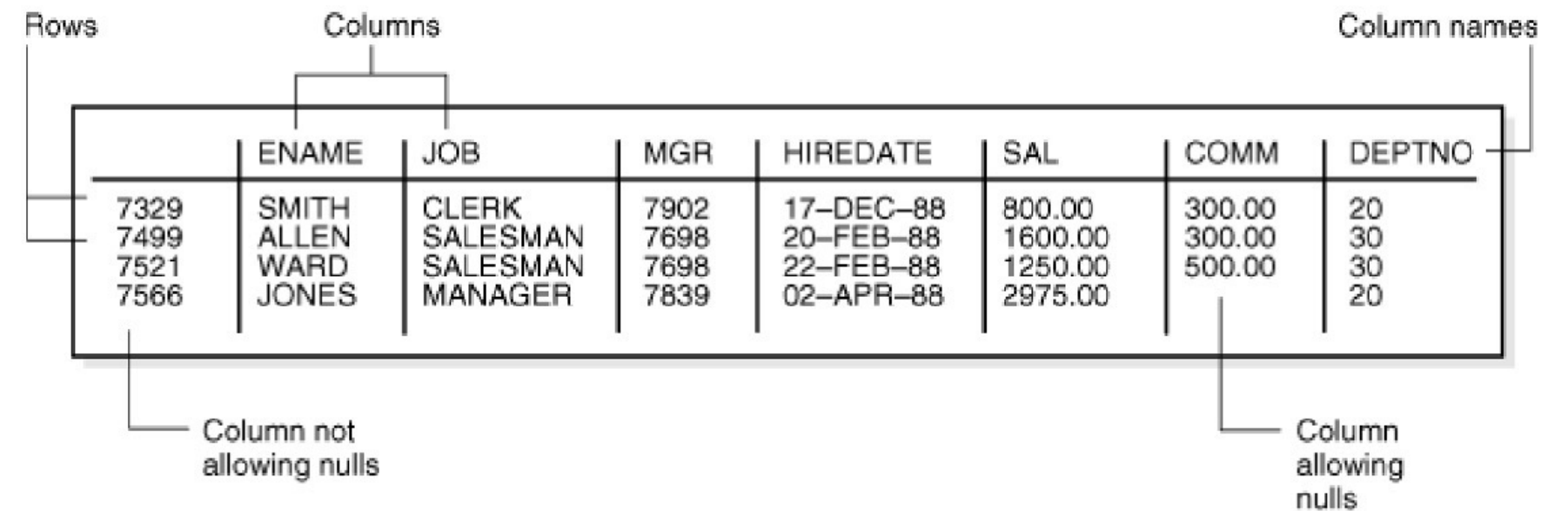
```
GRANT CONNECT, RESOURCE, CREATE VIEW, CREATE SEQUENCE TO uminho;
```



# Oracle Enterprise 19c

## ➔ Objects

- a. Tables
- b. Views
- c. Sequences
- d. Indexes
- e. Synonyms
- f. Procedures
- g. Functions
- (...)



The diagram illustrates a table structure with the following columns: ENAME, JOB, MGR, HIREDATE, SAL, COMM, and DEPTNO. The table contains four rows of data. Annotations include: 'Rows' pointing to the data rows, 'Columns' pointing to the column headers, 'Column names' pointing to the header row, 'Column not allowing nulls' pointing to the ENAME column, and 'Column allowing nulls' pointing to the COMM column.

	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7329	SMITH	CLERK	7902	17-DEC-88	800.00	300.00	20
7499	ALLEN	SALESMAN	7698	20-FEB-88	1600.00	300.00	30
7521	WARD	SALESMAN	7698	22-FEB-88	1250.00	500.00	30
7566	JONES	MANAGER	7839	02-APR-88	2975.00		20

# Oracle Enterprise 19c

## ➔ Physical Model

Exemplo de criação de tabela:

```
CREATE TABLE editor
(
    "id_editor" NUMBER(3, 0) NOT NULL ENABLE,
    "name" VARCHAR2(200 byte) NOT NULL ENABLE,
    CONSTRAINT "EDITOR_PK" PRIMARY KEY ("id_editor")
);

INSERT INTO editora (id_editor, name) VALUES (1, 'Penguin Books');
```

# Oracle Enterprise 19c

## ➔ Container

1. Enter the [Oracle Container Registry](#) and go to *database*;
  - a) Choose the *enterprise* repository
  - b) Create an Oracle account or log in
  - c) To pull the image, you'll need to authenticate first. To do this, run the following command and enter your Oracle credentials:

```
$ docker login container-registry.oracle.com
```

3. Pull the Oracle 19c image:

```
$ docker pull container-registry.oracle.com/database/enterprise:latest
```

# Oracle Enterprise 19c

## ➔ Container

4. Ensure Oracle image is available:

```
$ docker image ls
```

5. Follow the instructions on the Oracle Container Registry page to launch an Oracle Database instance. PS: set a password using the `-e ORACLE_PWD` parameter.

# NoSQL Databases

PL01 – Installations, Introduction to  
Docker, Introduction to Oracle

**Teacher:** Cristiana Neto

**Email:** [cristiana.neto@algoritmi.uminho.pt](mailto:cristiana.neto@algoritmi.uminho.pt)

**Office hours:**

Friday 10h–11h

