

# Exercícios Teóricos

## 1. (Conceito)

Explique o que é infraestrutura de TI e cite dois exemplos de como ela varia de acordo com o porte da organização.

## 2. (Data Center e PCs)

Quais são os principais fatores a serem considerados no dimensionamento de um Data Center e no dimensionamento de PCs para os usuários?

## 3. (Desenvolvimento Desktop vs. Web vs. Mobile)

Cite duas diferenças entre a infraestrutura necessária para o desenvolvimento desktop e a infraestrutura para desenvolvimento web.

## 4. (Cloud Computing)

Compare os modelos SaaS, PaaS e IaaS, destacando a principal responsabilidade do usuário em cada um.

## 5. (Gestão de Infraestrutura)

Liste três boas práticas de gestão de infraestrutura de TI que ajudam a reduzir falhas e melhorar o desempenho.

# Exercícios Práticos

## 6. (Escolha de SO)

Uma empresa de design gráfico precisa de alto desempenho em edição de imagens. Qual sistema operacional você recomendaria: Linux, Windows ou macOS? Justifique.

## 7. (Desenvolvimento Desktop em Java)

Liste três ferramentas necessárias para desenvolver um sistema desktop em Java.

## 8. (Arquitetura Web)

Um e-commerce será desenvolvido em PHP. Monte um esquema simples indicando: frontend, backend e banco de dados.

## 9. (Desenvolvimento Mobile)

Indique um framework híbrido para criar um app de delivery e explique por que ele é adequado.

## 10. (Cloud)

Um time de desenvolvimento quer implantar uma aplicação sem comprar servidores físicos. Qual modelo de serviço em nuvem deve usar e cite um provedor.

---

## Gabarito

1. Conjunto de hardware, software, redes e serviços que sustentam SI. Ex.: startup com notebooks e nuvem; banco com data centers robustos.
2. Data Center → servidores, energia, refrigeração. PCs → desempenho adequado às rotinas dos usuários.
3. Web exige servidor HTTP e arquitetura cliente-servidor; desktop precisa de IDE, compilador, SGBD local.
4. SaaS → usuário apenas usa o software; PaaS → usuário gerencia apps e desenvolvimento; IaaS → usuário gerencia SO e aplicativos.
5. Manutenção preventiva, monitoramento de desempenho, treinamentos para autonomia dos usuários.
6. macOS ou Windows (alta compatibilidade com softwares de design). macOS se destaca pela otimização gráfica.
7. Eclipse IDE, JDK, MySQL (ou outro SGBD).
8. Frontend: HTML/CSS/JS; Backend: PHP + Apache; Banco: MySQL.
9. React Native → permite criar apps híbridos para iOS/Android com JavaScript.
10. PaaS, ex.: Heroku, Google App Engine.