

Aula 1 mais detalhada:

✓ Respostas completas (Aula 01) – ECM251

📌 1. O que é Software?

Software é um conjunto de **programas, dados e instruções** que controlam o funcionamento de um computador ou sistema.

👉 Pode ser dividido em:

- **Software de sistema:** como sistemas operacionais (Windows, Linux)
 - **Software de aplicação:** como navegadores, editores de texto
 - **Software embarcado:** presente em dispositivos (ex: carros, geladeiras, satélites)
-

💻 2. Quem desenvolve Software?

Profissionais chamados de **desenvolvedores de software**, também conhecidos como:

- Programadores
- Engenheiros de Software
- Analistas de Sistemas
- Devs (front-end, back-end, fullstack etc.)

Esses profissionais trabalham de forma colaborativa em equipes, geralmente dentro de um processo de Engenharia de Software.

🏗️ 3. O que é Engenharia de Software?

É a área da computação que **aplica princípios de engenharia ao desenvolvimento de software**.


O objetivo é criar **software de qualidade**, que seja:

- Confiável

- Manutenível
- Escalável
- Entregue no prazo e dentro do orçamento

4. Por que a Engenharia de Software é importante?

- Reduz riscos de falhas no software
- Melhora a produtividade da equipe
- Garante organização e documentação
- Permite controle de versões e testes
- Facilita a manutenção e evolução do sistema
- Evita retrabalho e desperdício de recursos

 Em sistemas críticos (como os de aviões ou satélites), a falha de software pode causar prejuízos sérios ou até perdas humanas.

5. Quais são as etapas da Engenharia de Software?

Etapa	Descrição
1. Levantamento de Requisitos	O que o cliente precisa? (funcionalidades, restrições, metas)
2. Análise	Traduzir os requisitos em um modelo lógico
3. Projeto (Design)	Definir a arquitetura do sistema (módulos, classes, interfaces)
4. Implementação (Codificação)	Programar o sistema
5. Testes	Verificar se tudo funciona como esperado
6. Implantação	Colocar em produção para uso real
7. Manutenção	Corrigir erros e fazer melhorias contínuas

6. O que é um Artefato na Engenharia de Software?

É qualquer produto gerado durante o desenvolvimento de software, como:

- Diagramas (casos de uso, classes)
- Documentos de requisitos
- Código-fonte
- Casos de teste
- Manuais do usuário
- Relatórios de bugs

Tudo que documenta ou apoia o processo de desenvolvimento é considerado um artefato.

7. Como garantir que o trabalho foi realizado corretamente?

- **Revisões técnicas:** revisão entre pares, code review
- **Testes automatizados e manuais:** unitários, integração, sistema
- **Validação com o cliente:** testes de aceitação
- **Verificação com documentação:** tudo implementado de acordo com os requisitos?

A qualidade do software depende do processo, da comunicação, da validação e da documentação.

8. O que é Paradigma?

É uma **forma de pensar e estruturar soluções em programação**.

É como se fosse uma "filosofia de codar".

Exemplos:

- Paradigma Imperativo

- Paradigma Funcional
 - Paradigma Orientado a Objetos
-

9. O que é o Paradigma da Orientação a Objetos (OO)?

É um modelo de pensamento onde o mundo é representado por **objetos que interagem entre si**.

Cada objeto é uma instância de uma **classe**, e possui:

- **Atributos** (estado)
- **Métodos** (comportamentos)

Objetos compartilham mensagens para resolver problemas em conjunto. Esse paradigma facilita a **reutilização, modularidade, encapsulamento e manutenibilidade** do software.

10. O que é Programação Orientada a Objetos (POO)?

É a **implementação do paradigma OO** em linguagens como Java, Python, C++, etc.

Usa conceitos como:

- Classe e Objeto
- Encapsulamento
- Herança
- Polimorfismo
- Abstração

Na prática, programar com POO significa criar classes bem definidas, trabalhar com objetos e organizar o código para ser legível, reutilizável e escalável.
