NAME : FRIDASAM K. MURIITHI

TOPIC: TCHNICAL WRITING

1. **Understanding your audience’s expertise level** helps in adjusting language complexity, tone, and depth of explanation. Tech experts prefer concise, technical details, while general users need simplified, step-by-step guidance.
2. **Strategies to tailor content** include:
   * Using plain language for non-technical audiences
   * Providing analogies or real-life examples
   * Offering different formats (summaries, deep dives, FAQs)
   * Structuring content for skimming (headings, bullet points)
3. **Gauging audience knowledge** can be done by:
   * Surveys or feedback forms
   * Observing past interactions or questions
   * Checking user personas and demographics
   * Using progressive disclosure (start simple, add depth as needed)
4. **Ensuring accessibility** involves:
   * Avoiding jargon or defining it clearly
   * Using step-by-step explanations
   * Incorporating visuals and examples
   * Keeping sentences short and active
5. **Plain language** improves readability and ensures all users, including non-experts, can understand the content quickly without frustration.
6. **Examples of simplifying terms:**
   * *Start* instead of *Initiate*
   * *Use* instead of *Utilize*
   * *Check* instead of *Verify*
   * *Fix* instead of *Rectify*
7. **Examples and visuals** make concepts more concrete, reduce cognitive load, and improve retention by providing alternative ways to absorb information.
8. **Effective visuals for different technical content:**
   * **Diagrams** for system architecture or workflows
   * **Charts** for statistical data or comparisons
   * **Screenshots** for UI explanations
   * **Infographics** for high-level overviews
9. **Headings and subheadings** improve readability by breaking content into logical sections, making it easier to scan and navigate.
10. **Best practices for headings and subheadings:**
    * Use descriptive but concise titles
    * Follow a logical hierarchy (H1 → H2 → H3)
    * Keep them consistent and parallel in structure
    * Avoid vague titles like "Introduction"—be specific
11. **A README introduction** should immediately answer:
    * What the product does
    * Who it is for
    * Key features or benefits
    * Where to find installation and usage instructions
12. **Succinctly conveying purpose and key features:**
    * Use a one-liner description (e.g., *"X is a task management app that helps teams collaborate effortlessly."*)
    * List key features in bullet points
    * Highlight the problem it solves